9.4

Configuración de IBM MQ

Nota

Antes de utilizar esta información y el producto al que da soporte, lea la información en “Avisos” en la página 1079.

Esta edición se aplica a la versión 9 release 4 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.

Contenido

Configuración de IBM MQ

Cree uno o más gestores de colas en uno o varios sistemas y configúrelos en los sistemas de desarrollo, prueba y producción para procesar mensajes que contienen los datos de su empresa.

Acerca de esta tarea

Antes de configurar IBM MQ, lea los conceptos de IBM MQ en IBM MQ Visión general técnica. Lea sobre cómo planificar el entorno de IBM MQ en Planificación.

Existen una serie de métodos diferentes que puede utilizar para crear, configurar y administrar los gestores de colas y sus recursos relacionados en IBM MQ. Estos métodos incluyen interfaces de línea de mandatos, una interfaz gráfica de usuario y una API de administración. Si desea más información sobre estas interfaces, consulte Administración de IBM MQ.

Si desea instrucciones sobre cómo crear, iniciar, detener o suprimir un gestor de colas, consulte “Creación de gestores de colas en Multiplatforms” en la página 7.

Para obtener información sobre cómo crear los componentes necesarios para conectar entre sí las instalaciones y aplicaciones de IBM MQ, consulte “Configuración de la gestión de colas distribuidas” en la página 210.

Para obtener instrucciones sobre cómo conectar los clientes a un servidor de IBM MQ utilizando distintos métodos, consulte “Configuración de conexiones entre el cliente y el servidor” en la página 16.

Para obtener instrucciones sobre cómo configurar un clúster de gestor de colas, consulte “Configuración de un clúster de gestores de colas” en la página 309.

Puede cambiar el comportamiento de IBM MQ o un gestor de colas modificando la información de configuración. Para obtener más información, consulte “Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms” en la página 95. En general, no es necesario reiniciar un gestor de colas para que los cambios de configuración surtan efecto, excepto cuando se indique en esta documentación del producto.

z/OS Para obtener instrucciones sobre cómo configurar IBM MQ for z/OS, consulte “Configuring queue managers on z/OS” en la página 896.

Conceptos relacionados

Visión general técnica de IBM MQ

Tareas relacionadas

Administración de objetos de IBM MQ locales

Administración de objetos de IBM MQ remotos

IBM i Administración de IBMi

z/OS Administración de IBM MQ for z/OS

Planificación

z/OS Planificación del entorno de IBM MQ en z/OS

“Configuring queue managers on z/OS” en la página 896

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

Multi Creación de gestores de colas en Multiplatforms

Antes de poder utilizar mensajes y colas, debe crear e iniciar al menos un gestor de colas y los objetos asociados al mismo. Un gestor de colas gestiona los recursos que tiene asociados, en particular las colas que posee. Proporciona servicios de colocación en cola a las aplicaciones para llamadas y mandatos MQI (Message Queuing Interface) para crear, modificar, mostrar y suprimir objetos de IBM MQ.

Antes de empezar

Importante: IBM MQ no admite los nombres de máquina que contienen espacios. Si instala IBM MQ en un sistema con un nombre de máquina que contiene espacios, no puede crear ningún gestor de colas.

Para poder crear un gestor de colas, hay que tener en cuenta varios aspectos, especialmente en un entorno de producción. Trabaje con la siguiente lista de comprobación:

La instalación asociada con el gestor de colas
Para crear un gestor de colas, utilice el mandato de control de IBM MQ **crtmqm**. El mandato **crtmqm** asocia automáticamente un gestor de colas a la instalación desde la que se ha emitido el mandato **crtmqm**. Para los mandatos que se ejecutan en un gestor de colas, debe emitir el mandato desde la instalación asociada al gestor de colas. Puede cambiar la instalación asociada de un gestor de colas mediante el mandato setmqm. Tenga en cuenta que el instalador de Windows no añade el usuario que realiza la instalación al grupo mqm; para obtener más detalles, consulte Autorización para administrar IBM MQ en AIX, Linux®, and Windows.

Convenios de denominación
Utilice nombres en mayúsculas para que pueda comunicarse con los gestores de colas de todas las plataformas. Recuerde que los nombres se asignarán tal como los escriba. Para no tener que escribir demasiado, no utilice nombres que sean innecesariamente largos.

Especificación de un nombre de gestor de colas
Cuando cree un gestor de colas, asegúrese de que ningún otro gestor de colas tenga el mismo nombre en la red. Los nombres de los gestores de colas no se verifican cuando se crea el gestor de colas y, si no son exclusivos, no le permitirán que cree canales para la gestión de colas distribuidas. Asimismo, si utiliza la red para la mensajería de publicación/suscripción, las se asocian al nombre del gestor de colas que las ha creado. Si los gestores de colas de la jerarquía tienen el mismo nombre, es posible que las publicaciones no lleguen a los mismos.

Un método para garantizar su exclusividad es poner delante de cada nombre de gestor de colas su propio nombre de nodo exclusivo. Por ejemplo, si se llama a un nodo ACCOUNTS, puede nombrar el gestor de colas ACCOUNTS.SATURN.QUEUE.MANAGER, donde SATURN identifica un gestor de colas determinado y QUEUE.MANAGER es una extensión que puede dar a todos los gestores de colas. De forma alternativa, puede omitir esto, pero tenga en cuenta que ACCOUNTS.SATURN y ACCOUNTS.SATURN.QUEUE.MANAGER son nombres de gestores de colas diferentes.

Si utiliza IBM MQ para la comunicación con otras empresas, también puede incluir su propio nombre de empresa como prefijo. Esto no se muestra en los ejemplos, porque dificulta su comprensión.

Nota: Los nombres de gestores de colas en los mandatos de control son sensibles a las mayúsculas y minúsculas. Esto significa que puede crear dos gestores de colas con los nombres jupiter.queue.manager y JUPITER.queue.manager. Sin embargo, es mejor evitar este tipo de complicaciones.

Limitación del número de gestores de colas
Puede crear tantos gestores de cola como permitan los recursos. Sin embargo, dado que cada gestor de colas requiere sus propios recursos, normalmente es mejor tener un gestor de colas con 100 colas en un nodo que diez gestores de colas con diez colas cada uno.

En los sistemas de producción, muchos procesadores se pueden utilizar con un solo gestor de colas, pero las máquinas servidor más grandes se ejecutan de forma más eficaz con varios gestores de colas.

Especificación de un gestor de colas predeterminado
Cada nodo debe tener un gestor de colas predeterminado, aunque es posible configurar IBM MQ en un nodo sin uno. El gestor de colas predeterminado es el gestor de colas al que se conectan las aplicaciones si éstas no especifican un nombre de gestor de colas en una llamada MQCONN. Es también el gestor de colas que procesa los mandatos MQSC cuando se invoca el mandato runmqsc sin especificar un nombre de gestor de colas.

Si se especifica un gestor de colas como predeterminado, sustituye cualquier especificación de gestor de colas predeterminado existente en el nodo.

Si se cambia el gestor de colas predeterminado, otros usuarios o aplicaciones pueden verse afectados. El cambio no afectará a las aplicaciones conectadas actualmente, ya que pueden utilizar el manejador de su llamada de conexión original en todas las llamadas MQI posteriores. Este manejador asegura que las llamadas se dirijan al mismo gestor de colas. Todas las aplicaciones que se conecten *después* de haber cambiado el gestor de colas predeterminado se conectarán al nuevo gestor de colas predeterminado. Quizá esta haya sido su intención, pero debe tenerlo en cuenta antes de cambiar el valor predeterminado.

En el apartado "Creación de un gestor de colas predeterminado" en la página 12 se explica cómo se crea un gestor de colas predeterminado.

Especificación de una cola de mensajes no entregados

La cola de mensajes no entregados es una cola local a la que se transfieren los mensajes que no se pueden direccionar a su destino correcto.

Es de vital importancia tener una cola de mensajes no entregados en cada gestor de colas de la red. Si no define una, los errores que se produzcan en los programas de aplicación pueden hacer que se cierren los canales y que no se reciban las respuestas a los mandatos de administración.

Por ejemplo, si una aplicación intenta transferir un mensaje a una cola de otro gestor de colas, pero se especifica un nombre de cola incorrecto, el canal se detiene y el mensaje permanece en la cola de transmisión. Otras aplicaciones no podrán utilizar este canal para sus mensajes.

Los canales no se ven afectados si los gestores de colas tienen colas de mensajes no entregados. El mensaje no entregado se transfiere a la cola de mensajes no entregados, en el extremo receptor, dejando disponibles el canal y su cola de transmisión.

Cuando cree un gestor de colas, utilice el distintivo **-u** para especificar el nombre de la cola de mensajes no entregados. También puede utilizar un mandato MQSC para modificar los atributos de un gestor de colas que ya esté definido y especificar la cola de mensajes no entregados que desea utilizar. Consulte Visualización y modificación de atributos del gestor de colas para ver un ejemplo del mandato de MQSC ALTER.

Especificación de una cola de transmisión predeterminada

Una cola de transmisión es una cola local en la que se colocan hasta su transmisión los mensajes en tránsito a un gestor de colas remoto. La cola de transmisión predeterminada es la cola que se utiliza cuando no se define explícitamente ninguna cola de transmisión. Se puede asignar una cola de transmisión predeterminada a cada gestor de colas.

Cuando cree un gestor de colas, utilice el distintivo **-d** para especificar el nombre de la cola de transmisión predeterminada. Con esto no se crea en realidad la cola; tiene que hacerlo explícitamente más adelante. Consulte Trabajar con colas locales si desea más información.

Especificación de los parámetros de anotaciones cronológicas necesarios

Puede especificar parámetros de anotaciones cronológicas en el mandato `crtmqm`, incluidos el tipo de anotaciones, la vía de acceso y el tamaño de los archivos de anotaciones.

En un entorno de desarrollo, los parámetros de registro cronológico predeterminados deberían ser los adecuados. No obstante, puede cambiar los valores predeterminados si, por ejemplo:

- Tiene una configuración del sistema de gama baja que no puede dar soporte a registros grandes.
- Prevé que habrá un gran número de mensajes largos en las colas simultáneamente.
- Cree que se transferirán muchos mensajes persistentes a través del gestor de colas.

Cuando haya establecido los parámetros de registro cronológico, solamente podrá modificar algunos de ellos si suprime el gestor de colas y vuelve a crearlo con el mismo nombre pero con diferentes parámetros de registro cronológico.

Si desea obtener más información acerca de los parámetros de registro cronológico, consulte "Registro: Asegurarse de que no se han perdido mensajes" en la página 673.

AIX **Sólo para sistemas IBM MQ for UNIX**

Puede crear el directorio del gestor de colas `/var/mqm/qmgrs/qmgr`, incluso en un sistema de archivos local distinto, antes de utilizar el mandato **crtmqm**. Cuando utilice **crtmqm**, si el directorio `/var/mqm/qmgrs/qmgr` existe, está vacío y su propietario es mqm, se utilizará para los datos del gestor de colas. Si el directorio no es propiedad de mqm, la creación falla con un mensaje de First Failure Support Technology (FFST). Si el directorio no está vacío, se crea un nuevo directorio.

Acerca de esta tarea

Para crear un gestor de colas, utilice el mandato de control de IBM MQ **crtmqm**. Para obtener más información, consulte **crtmqm**. El mandato **crtmqm** crea automáticamente los objetos predeterminados y los objetos del sistema necesarios (consulte Objetos predeterminados del sistema). Los objetos predeterminados forman la base de todas las definiciones de objeto que efectúe; los objetos del sistema son necesarios para el funcionamiento del gestor de colas.

Windows En sistemas Windows se tiene la opción de iniciar varias instancias del gestor de colas utilizando la opción *sax* del mandato **crtmqm**.

Cuando haya creado un gestor de colas y sus objetos, se puede usar el mandato **strmqm** para iniciar el gestor de colas.

Procedimiento

- Para obtener información de ayuda en la creación y gestión de gestores de colas, consulte los subtemas siguientes:
  - “Creación de un gestor de colas predeterminado” en la página 12
  - “Convertir un gestor de colas existente en el gestor de colas predeterminado” en la página 13
  - “Copia de seguridad de los archivos de configuración después de crear un gestor de colas” en la página 14

Conceptos relacionados

Trabajar con gestores de colas

Tareas relacionadas

Creación de un gestor de colas llamado QM1

“Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms” en la página 95

Puede cambiar el comportamiento de IBM MQ o de un gestor de colas individual para que se ajuste a las necesidades de la instalación editando la información en los archivos de configuración (`.ini`). También puede cambiar las opciones de configuración para IBM MQ MQI clients.

“Configuring queue managers on z/OS” en la página 896

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

Referencia relacionada

Objetos predeterminados y del sistema

crtmqm

Linux AIX Directorio efímero configurable

El directorio efímero configurable define la ubicación a la que deberían ir los datos efímeros en el gestor de colas. Esto se puede utilizar para permitir que los sockets del dominio AIX and Linux se coloque en un sistema de archivos no montado en un entorno de Red Hat® OpenShift®.

Con anterioridad a IBM MQ 9.2.0, en las plataformas AIX and Linux cuando se ejecuta un gestor de colas, los sockets de dominio de AIX and Linux se crean bajo el directorio `/var/mqm/sockets`. Al ejecutar el gestor de colas dentro de un contenedor, con `/var/mqm` como sistema de archivos montado, algunas plataformas Linux pueden impedir la creación de estos sockets de dominio, porque permiten que algunos

procesos de fuera del contenedor interfieran con las operaciones dentro del contenedor. Este problema impide que IBM MQ se ejecute en una plataforma de contenedor de Red Hat OpenShift, bajo el contexto de seguridad predeterminado.

Desde IBM MQ 9.2.0, el atributo **EphemeralPrefix** se puede utilizar para configurar la ubicación del directorio efímero. Si no utiliza este atributo, no verá ningún cambio en el comportamiento.

Cuando se crea una entrada de gestor de colas en `mqs.ini` (utilizando los mandatos **crtmqm** o **addmqinf**), se añade el atributo **EphemeralPrefix** si:

- Establezca el atributo **DefaultEphemeralPrefix** en "Stanza AllQueueManagers del archivo mqs.ini" en la página 101.
- Establezca la variable de entorno **MQ_EPHEMERAL_PREFIX** .
- Especifica **-v** `EphemeralPrefix` solo para el mandato **addmqinf**.

También puede añadir de forma explícita el atributo **EphemeralPrefix** a un gestor de colas existente cuando se detiene, y este se añade cuando se reinicia el gestor de colas.

Si especifica un atributo **EphemeralPrefix**, cuando se inicia el gestor de colas, hace que los datos efímeros para el gestor de colas se creen bajo ese prefijo, en lugar de en su ubicación habitual. Es decir:

- Los archivos de socket que normalmente están presentes en `/var/mqm/sockets/<QM>` estarán ahora en `/<EphemeralPrefix>/sockets/<QM>`
- Los archivos de subagrupación que normalmente están presentes en `/<Prefix>/qmgrs/<QM>/@<Subpool>` estarán ahora en `/<EphemeralPrefix>/qmgrs/<QM>/@<Subpool>`

Notas:

- `/var/mqm/sockets/@SYSTEM` permanece en su ubicación fija y no forma parte del atributo **EphemeralPrefix**.
- `AMQCLCHL.TAB` permanece bajo `/<Prefix>/qmgrs/<QM>/@ipcc` y no forma parte del atributo **EphemeralPrefix** .

El número de caracteres que el atributo **EphemeralPrefix** puede incluir depende de su plataforma:

- Linux AIX En plataformas AIX and Linux está limitado a 12 caracteres.
- IBM i En IBM i está limitado a 24 caracteres.

Si especifica un atributo **EphemeralPrefix** que es demasiado largo, o no existe, recibirá un mensaje AMQ7001E:

```
AMQ7001E: La ubicación especificada para el gestor de colas no es válida
```

.

Multi Directorio userdata

Puede utilizar el directorio `userdata` para almacenar el estado de la aplicación persistente.

Cada gestor de colas de IBM MQ tiene un sistema de archivos dedicado para su estado persistente, que incluye tanto sus datos de cola como el registro de recuperación. El sistema de archivos incluye un directorio `userdata` que puede utilizar para almacenar información de estado persistente para las aplicaciones. Consulte Contenido de directorios en sistemas Unix y Linux y Contenido de directorios en sistemas Windows.

El directorio `userdata` puede ser útil en varias situaciones, por ejemplo:

- En las configuraciones de RDQM, de modo que la información de la aplicación también se traslada cuando se produce una migración tras error del gestor de colas a otro nodo (consulte "Almacenamiento del estado de aplicación persistente" en la página 613).
- Para los gestores de colas multiinstancia, el estado de la aplicación se encuentra con sus datos de gestor de colas en el sistema de archivos de red compartido.
- De forma más general, donde las aplicaciones son servicios de gestor de colas configurados.

Si elige almacenar el estado de aplicación en el directorio `userdata`, debe tener en cuenta que los datos escritos en esta ubicación pueden consumir el espacio de disco disponible asignado al gestor de colas. Debe asegurarse de que haya suficiente espacio de disco disponible para que el gestor de colas escriba datos de cola, registros y otra información de estado persistente.

El directorio `userdata` tiene la propiedad de usuario y grupo mqm y lo puede leer todo el mundo para que los usuarios puedan acceder al mismo sin necesidad de pertenecer al grupo de administradores de IBM MQ (es decir, mqm). No puede modificar los permisos del directorio `userdata`, pero puede crear contenido en él con la propiedad y los permisos necesarios.

Multi Creación de un gestor de colas predeterminado

El gestor de colas predeterminado es el gestor de colas al que se conectan las aplicaciones si estas no especifican un nombre de gestor de colas en una llamada MQCONN. Es también el gestor de colas que procesa los mandatos MQSC cuando se invoca el mandato **runmqsc** sin especificar un nombre de gestor de colas. Para crear un gestor de colas, utilice el mandato de control de IBM MQ **crtmqm**.

Antes de empezar

Antes de crear un gestor de colas predeterminado, lea las consideraciones descritas en "Creación de gestores de colas en Multiplatforms" en la página 7.

Linux AIX Cuando se utiliza **crtmqm** para crear un gestor de colas en AIX and Linux, si el directorio `/var/mqm/qmgrs/qmgr` ya existe, es propiedad de mqm y está vacío, se utiliza para los datos del gestor de colas. Si el directorio no es propiedad de mqm, la creación del gestor de colas falla con un mensaje de First Failure Support Technology (FFST). Si el directorio no está vacío, se crea un nuevo directorio para los datos del gestor de colas.

Esta consideración se aplica incluso cuando el directorio `/var/mqm/qmgrs/qmgr` ya existe en un sistema de archivos local independiente.

Acerca de esta tarea

Cuando se crea un gestor de colas con el mandato **crtmqm**, este crea automáticamente los objetos predeterminados y de sistema necesarios. Los objetos predeterminados forman la base de todas las definiciones de objeto realizadas y los objetos del sistema son necesarios para el funcionamiento del gestor de colas.

Incluyendo los correspondientes parámetros en el mandato, también se puede definir, por ejemplo, el nombre de la cola de transmisión predeterminada usada por el gestor de colas y el nombre de la cola de mensajes no entregados.

Windows En Windows, se puede usar la opción **sax** del mandato **crtmqm** para iniciar varias instancias del gestor de colas.

Para obtener más información sobre el mandato **crtmqm** y su sintaxis, consulte **crtmqm**.

Procedimiento

- Para crear un gestor de colas predeterminado, utilice el mandato **crtmqm** con el distintivo **-q**.
  El ejemplo siguiente del mandato **crtmqm** crea un gestor de colas predeterminado llamado SATURN.QUEUE.MANAGER:

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

  donde:

  -q
  Indica que este gestor de colas es el gestor de colas predeterminado.

-d MY.DEFAULT.XMIT.QUEUE
Es el nombre de la cola de transmisión predeterminada que ha de utilizar este gestor de colas.

Nota: IBM MQ no crea una cola de transmisión predeterminada automáticamente; debe definirla usted mismo.

-u SYSTEM.DEAD.LETTER.QUEUE
Es el nombre de la cola de mensajes no entregados predeterminada creada por IBM MQ en la instalación.

SATURN.QUEUE.MANAGER
Es el nombre de este gestor de colas. Ha de ser el último parámetro especificado en el mandato `crtmqm`.

Qué hacer a continuación

Cuando haya creado un gestor de colas y sus objetos, utilice el mandato **strmqm** para Iniciar el gestor de colas.

Conceptos relacionados

Trabajar con colas locales

Tareas relacionadas

"Copia de seguridad de los archivos de configuración después de crear un gestor de colas" en la página 14

La información de configuración de IBM MQ se almacena en los archivos de configuración en AIX, Linux, and Windows. Después de crear un gestor de colas, haga una copia de seguridad de los archivos de configuración. A continuación, si crea otro gestor de colas que le causa algún problema, puede reinstalar las copias de seguridad cuando haya eliminado la causa del problema.

Visualización y modificación de atributos del gestor de colas

Referencia relacionada

Objetos predeterminados y del sistema

Multi Convertir un gestor de colas existente en el gestor de colas predeterminado

Puede convertir un gestor de colas existente en el gestor de colas predeterminado manualmente utilizando un editor de texto o, en Windows y Linux, utilizando IBM MQ Explorer.

Acerca de esta tarea

Para utilizar un editor de texto para convertir un gestor de colas existente en el gestor de colas predeterminado, realice los pasos siguientes.

Windows Linux En sistemas Windows y Linux (plataformas x86 y x86-64), si prefiere utilizar IBM MQ Explorer para realizar este cambio, consulte "Utilización de IBM MQ Explorer para convertir a un gestor de colas en el valor predeterminado" en la página 14.

Cuando crea un gestor de colas predeterminado, su nombre se inserta en el atributo `Name` de la stanza `DefaultQueueManager` del archivo de configuración IBM MQ (`mqs.ini`). La stanza y su contenido se crean automáticamente si no existen.

Procedimiento

- Para convertir un gestor de colas existente en el gestor de colas predeterminado, cambie el nombre del gestor de colas que aparece en el atributo `Name` por el nombre del nuevo gestor de colas predeterminado. Puede hacer esto manualmente, con un editor de texto.

- Si no tiene un gestor de colas predeterminado en el nodo y desea convertir un gestor de colas existente en el predeterminado, cree usted mismo la stanza *DefaultQueueManager* con el nombre necesario.
- Si accidentalmente convierte otro gestor de colas en el valor predeterminado y desea revertir al gestor de colas predeterminado original, edite la stanza `DefaultQueueManager` en `mqs.ini`, sustituyendo el gestor de colas predeterminado no deseado por el que desea.

Tareas relacionadas

"Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms" en la página 95

Puede cambiar el comportamiento de IBM MQ o de un gestor de colas individual para que se ajuste a las necesidades de la instalación editando la información en los archivos de configuración (`.ini`). También puede cambiar las opciones de configuración para IBM MQ MQI clients.

Windows Linux Utilización de IBM MQ Explorer para convertir a un gestor de colas en el valor predeterminado

En sistemas Windows y Linux (plataformas x86 y x86-64), puede utilizar IBM MQ Explorer para convertir a un gestor de colas existente en el gestor de colas predeterminado.

Acerca de esta tarea

Para utilizar IBM MQ Explorer para convertir a un gestor de colas existente en el gestor de colas predeterminado en sistemas Windows y Linux (plataformas x86 y x86-64), complete los siguientes pasos.

Si prefiere utilizar un editor de texto para realizar este cambio manualmente, consulte "Convertir un gestor de colas existente en el gestor de colas predeterminado" en la página 13.

Procedimiento

1. Abra IBM MQ Explorer.
2. Pulse el botón derecho del ratón en **IBM MQ** y, a continuación, seleccione **Propiedades ...**. Se visualiza el panel **Propiedades de IBM MQ**.
3. Escriba el nombre del gestor de colas predeterminado en el campo **Nombre de gestor de colas predeterminado**.
4. Pulse **Aceptar**.

ALW Copia de seguridad de los archivos de configuración después de crear un gestor de colas

La información de configuración de IBM MQ se almacena en los archivos de configuración en AIX, Linux, and Windows. Después de crear un gestor de colas, haga una copia de seguridad de los archivos de configuración. A continuación, si crea otro gestor de colas que le causa algún problema, puede reinstalar las copias de seguridad cuando haya eliminado la causa del problema.

Acerca de esta tarea

Por regla general, debería hacer una copia de seguridad de los archivos de configuración cada vez que cree un nuevo gestor de colas.

Hay dos tipos de archivos de configuración:

- Al instalar el producto, se crea el archivo de configuración IBM MQ (`mqs.ini`). Este archivo contiene una lista de gestores de colas, que se actualiza cada vez que se crea o suprime un gestor de colas. Hay un archivo `mqs.ini` por nodo.
- Al crear un nuevo gestor de colas, se crea automáticamente un nuevo archivo de configuración de gestor de colas (`qm.ini`). Este archivo contiene parámetros de configuración del gestor de colas.

Si ha instalado el servicio AMQP, hay un archivo de configuración adicional del que debe realizar copia de seguridad:

- Windows En sistemas Windows: `amqp_win.properties`
- Linux AIX En sistemas AIX and Linux: `amqp_unix.properties`

Tareas relacionadas

“Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms” en la página 95
Puede cambiar el comportamiento de IBM MQ o de un gestor de colas individual para que se ajuste a las necesidades de la instalación editando la información en los archivos de configuración (`.ini`). También puede cambiar las opciones de configuración para IBM MQ MQI clients.

“Copia de seguridad y restauración de datos de gestor de colas de IBM MQ” en la página 706
Se pueden proteger los gestores de colas frente a posibles corrupciones producidas por fallos de hardware haciendo copias de seguridad de los gestores de colas y de sus datos, haciendo una copia de seguridad solo de la configuración del gestor de colas y usando un gestor de colas de copia de seguridad.

Traslado de un gestor de colas a un sistema operativo diferente

Siga estas instrucciones para mover un gestor de colas de un sistema operativo a otro. Tenga en cuenta que esto es **no** una migración de un gestor de colas.

Acerca de esta tarea

Un gestor de colas se mueve creándolo de nuevo en el sistema de destino. El procedimiento vuelve a crear la configuración del gestor de colas, no intenta crear de nuevo el estado actual del mismo descargando y volviendo a cargar las colas, por ejemplo.

Procedimiento

1. Inicie sesión en el sistema de origen con un usuario del grupo de administradores de IBM MQ (mqm).
2. Guarde la información de configuración del gestor de colas que desee mover escribiendo el comando siguiente:

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

   Donde:

   - *nombre_QM* es el nombre del gestor de colas que desea mover.
   - *archivo_QM* es el nombre y la vía de acceso del archivo local en el sistema de origen en el que se escribe la información de configuración.

   Consulte **dmpmqcfg** si desea más información.
3. Si el gestor de colas forma parte de una configuración distribuida, desactive temporalmente el gestor de colas. Asegúrese de que no haya mensajes en curso y pare el gestor de colas.
4. Si está moviendo de una versión del producto a otra, migre el gestor de colas del sistema operativo actual a la última versión.

   Si el sistema operativo actual es:

   - Windows Windows, consulte Migración de un gestor de colas en Windows
   - Linux AIX AIX o Linux, consulte Migración de un gestor de colas en AIX and Linux

   Hay que asegurarse de que las aplicaciones existentes sigan funcionando.
5. Cree un gestor de colas vacío en el nuevo sistema operativo utilizando **crtmqm**.
6. Copie las definiciones de objeto en el gestor de colas recién creado utilizando **dmpmqcfg**.

Tenga mucho cuidado al copiar las definiciones de objeto, porque podría ser necesario modificar manualmente las definiciones:

- Hay que comprobar varios atributos por si fuera necesario modificarlos. Incluyen los siguientes:
  - Direcciones IP y puertos de canales, escuchas y otros objetos
  - Información de seguridad como, por ejemplo, identificadores de usuario
  - **`startcmd`** en servicios
  - Otros atributos.
- Los suscriptores duraderos que no son administrados podrían perder mensajes.
- También podría ser necesario cambiar otros gestores de colas para que sus canales se conecten con el gestor de colas movido.

Una vez copiadas las definiciones, hay que copiar los mensajes de aplicación del gestor de colas del sistema operativo original al gestor de colas del nuevo sistema operativo, utilizando una aplicación que mueva los mensajes. Luego, compruebe que las aplicaciones siguen funcionando.

Configuración de conexiones entre el cliente y el servidor

Para configurar los enlaces de comunicación entre IBM MQ MQI clients y servidores, decida el protocolo de comunicación, defina las conexiones en ambos extremos del enlace, inicie un escucha y defina canales.

Acerca de esta tarea

En IBM MQ, los enlaces de comunicación lógicos entre objetos se denominan *canales*. Los canales utilizados para conectar IBM MQ MQI clients con servidores se denominan canales MQI. Debe configurar las definiciones de canal en cada extremo del enlace para que la aplicación IBM MQ en el IBM MQ MQI client se pueda comunicar con el gestor de colas en el servidor.

Antes de definir los canales MQI, hay que decidir qué forma de comunicación se va a utilizar y definir la conexión en cada extremo del canal.

Si está definiendo un canal MQI entre un IBM MQ MQI client y un gestor de colas que están en redes físicas diferentes, o que se comunican a través de un cortafuegos, el uso de IBM MQ Internet Pass-Thru podría simplificar la configuración. Para obtener más información, consulte IBM MQ Internet Pass-Thru.

Procedimiento

1. Decida la forma de comunicación que va a utilizar.
   Consulte “Qué tipo de comunicación utilizar” en la página 17.
2. Defina la conexión en cada extremo del canal.
   Para definir la conexión, debe:
   a) Configurar la conexión
   b) Anotar los valores de los parámetros que necesita para las definiciones de canal.
   c) Habilitar el servidor para que detecte las solicitudes de red entrantes del IBM MQ MQI client, iniciando un *escucha*.

Conceptos relacionados

“Archivo de configuración de IBM MQ MQI client , mqclient.ini” en la página 172
Puede configurar los clientes utilizando atributos en un archivo de texto. Estos atributos se pueden alterar temporalmente con variables de entorno o de otras formas según la plataforma específica.

Tareas relacionadas

“Utilización de las variables de entorno de IBM MQ” en la página 66
Puede utilizar mandatos para visualizar los valores actuales o restablecer los valores de las variables de entorno de IBM MQ.

Conexión de aplicaciones de cliente MQI de IBM MQ con gestores de colas

Referencia relacionada

DISPLAY CHLAUTH

SET CHLAUTH

Qué tipo de comunicación utilizar

Diferentes plataformas dan soporte a diferentes protocolos de comunicación. El protocolo de transmisión que elija dependerá de su combinación de plataformas de servidor y IBM MQ MQI client.

Tipos de protocolo de transmisión en canales MQI

Dependiendo de las plataformas servidora y cliente, existen hasta cuatro tipos de protocolo de transmisión en un canal MQI:

- TCP/IP
- LU6.2
- NetBIOS
- SPX

Cuando se definen los canales MQI, cada definición de canal debe especificar un atributo de protocolo de transmisión (tipo de transporte). Un servidor no está restringido a un protocolo, por lo que distintas definiciones de canal pueden especificar protocolos diferentes. Para IBM MQ MQI clients, podría ser conveniente tener canales MQI alternativos que utilicen protocolos de transmisión diferentes.

La elección del protocolo de transmisión también depende de la combinación específica de plataformas de cliente y servidor de IBM MQ. Las combinaciones posibles se muestran en la tabla siguiente.

Tabla 1. Protocolos de transmisión - combinación de plataformas de servidor y IBM MQ MQI client

Protocolo de transmisión	**IBM MQ MQI client**	**Servidor de IBM MQ**
TCP/IP "1" en la página 18	IBM i IBM i AIX AIX Linux Linux Windows Windows	IBM i IBM i AIX AIX Linux Linux Windows Windows z/OS z/OS
LU6.2	AIX AIX Linux Linux"2" en la página 18 Windows Windows	IBM i IBM i AIX AIX Linux Linux"2" en la página 18 Windows Windows z/OS z/OS
NetBIOS	Windows Windows	Windows Windows
SPX	Windows Windows	Windows Windows

Notas:

1. 

Un canal de mensajes que utiliza TCP/IP puede apuntar a un IBM Aspera faspio Gateway, que proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de la red. Consulte Definición de una conexión de Aspera gateway en Linux o Windows.
2. Excepto Linux (plataforma POWER)

Conceptos relacionados

"Definición de una conexión TCP en Windows" en la página 274
Defina una conexión TCP configurando un canal en el extremo emisor para especificar la dirección del destino y ejecutando un programa de escucha en el extremo receptor.

"Definición de una conexión TCP en AIX and Linux" en la página 282
La definición de canal en el extremo emisor especifica la dirección del destino. El daemon de escucha o inet está configurado para la conexión en el extremo receptor.

"Definición de una conexión TCP en IBM i" en la página 302
Puede definir una conexión TCP dentro de la definición de canal utilizando el campo Nombre de conexión.

"Defining a TCP connection on z/OS" en la página 996
To define a TCP connection, there are a number of settings to configure.

"Definición de una conexión LU 6.2 en Windows" en la página 276
SNA debe configurarse de manera que pueda establecerse una conversación LU 6.2 entre las dos máquinas.

"Definición de una conexión LU 6.2 en AIX and Linux" en la página 286
SNA debe configurarse de manera que pueda establecerse una conversación LU 6.2 entre las dos máquinas.

"Definición de una conexión LU 6.2 en IBM i" en la página 304
Defina los detalles de las comunicaciones LU 6.2 utilizando un nombre de modalidad, nombre de TP y el nombre de una conexión LU 6.2 totalmente calificada.

"Definición de una conexión NetBIOS en Windows" en la página 278
Una conexión NetBIOS se aplica únicamente a un cliente y un servidor que ejecuten Windows. IBM MQ utiliza tres tipos de recursos NetBIOS al establecer una conexión NetBIOS con otro producto IBM MQ: sesiones, mandatos y nombres. Cada uno de estos recursos tiene un límite, que se establece ya sea de forma predeterminada o por elección propia durante la instalación de NetBIOS.

Tareas relacionadas

"Definición de una conexión de Aspera gateway en plataformas Linux o Windows" en la página 887
El IBM Aspera faspio Gateway proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de red para IBM MQ. Un gestor de colas que se ejecuta en cualquier plataforma autorizada puede conectarse a través de un Aspera gateway. La propia pasarela se despliega en Red Hat o Ubuntu Linux, o Windows.

Referencia relacionada

"Límites de la conexión TCP/IP" en la página 18
El número de solicitudes de conexión pendientes que pueden colocarse en cola en un solo puerto TCP/IP depende de la plataforma. Se produce un error si se alcanza el límite.

"Defining an LU6.2 connection for z/OS using APPC/MVS" en la página 999
To define an LU6.2 connection there are a number of settings to configure.

Límites de la conexión TCP/IP

El número de solicitudes de conexión pendientes que pueden colocarse en cola en un solo puerto TCP/IP depende de la plataforma. Se produce un error si se alcanza el límite.

Este límite de conexiones no es lo mismo que el número máximo de clientes que se pueden conectar a un servidor de IBM MQ. Puede conectar más clientes a un servidor, hasta el nivel determinado por los recursos del sistema de servidor. Los valores de reserva de las solicitudes de conexión se muestran en la tabla siguiente:

Tabla 2. Número máximo de solicitudes de conexión pendientes puestas en cola en un puerto TCP/IP	
Plataforma de servidor	**Número máximo de solicitudes de conexión**
AIX	100
Linux	100
IBM i	255
Servidor de Windows	100
Estación de trabajo de Windows	100
z/OS	255

Si se alcanza el límite de conexión, el cliente recibe un código de retorno de MQRC_HOST_NOT_AVAILABLE de la llamada de `MQCONN` y un error AMQ9202 en el registro de errores del cliente (`/var/mqm/errors/AMQERR0n.LOG` en sistemas AIX and Linux o `amqerr0n.log` en el subdirectorio de errores de la instalación del cliente de IBM MQ en Windows). Si el cliente reintenta la petición `MQCONN`, es posible que se ejecute correctamente.

Para aumentar el número de solicitudes de conexión que puede efectuar y evitar que se generen mensajes de error debido a esta limitación, puede tener varios escuchas a la escucha cada uno en un puerto distinto o bien tener más de un gestor de colas.

Cómo configurar un IBM MQ MQI client

Para configurar un cliente, siga estas instrucciones.

Antes de empezar

Para configurar un IBM MQ MQI client debe tener un servidor IBM MQ ya instalado y funcionando, al que se conectará el cliente.

Procedimiento

1. Compruebe que tiene una plataforma adecuada para un cliente MQI de IBM MQ y que el hardware y el software cumplen los requisitos.

   El soporte de plataforma se describe en Soporte de plataforma para clientes IBM MQ.
2. Decida cómo va a instalar IBM MQ en la estación de trabajo cliente y, a continuación, siga las instrucciones para su combinación concreta de plataformas de cliente y servidor.

   La instalación se describe en los temas siguientes:

   - AIX Instalación de un cliente de IBM MQ en AIX
   - Linux Instalación de un cliente IBM MQ en Linux
   - Windows Instalación de un cliente IBM MQ en Windows
   - IBM i Instalación de un cliente IBM MQ en IBM i
3. Asegúrese de que los enlaces de comunicaciones están configurados y conectados.

   La configuración de los enlaces de comunicaciones se describe en Configuración de conexiones entre el servidor y el cliente.
4. Compruebe si la instalación funciona correctamente.

Consulte la sección de verificación del procedimiento de instalación para la o las plataformas que utilice su empresa.

5. Cuando tenga la instalación de IBM MQ MQI client verificada, tenga en cuenta si debe proteger el cliente.

   La seguridad del cliente se describe en Configuración de la seguridad de IBM MQ MQI client.

6. Configure los canales entre el cliente MQI de IBM MQ y el servidor que son necesarios para las aplicaciones IBM MQ que desea ejecutar en el cliente.

   La configuración de canales se describe en Definición de canales MQI. Hay algunas consideraciones adicionales si utiliza TLS.

   Estas consideraciones se describen en Especificación de que un canal MQI utiliza TLS. Es posible que tenga que utilizar un archivo de configuración IBM MQ MQI client o variables de entorno de IBM MQ para configurar los canales. Las variables de entorno de IBM MQ se describen en Utilización de las variables de entorno de IBM MQ.

7. Consulte Desarrollo de aplicaciones para obtener una descripción completa de las aplicaciones IBM MQ .

8. Al diseñar, crear y ejecutar aplicaciones en el entorno de IBM MQ MQI client , debe tener en cuenta las diferencias de un entorno de gestor de colas.

   Para obtener información sobre estas diferencias, consulte:

   - Utilización de la interfaz de cola de mensajes (MQI) en una aplicación cliente
   - Creación de aplicaciones para IBM MQ MQI clients
   - Conexión de aplicaciones IBM MQ MQI client a gestores de colas
   - Resolución de problemas con IBM MQ MQI clients

Configuración de un cliente transaccional extendido

En este colección de temas se describe cómo configurar la función transaccional extendida para cada categoría de gestor de transacciones.

Para cada plataforma, el cliente transaccional extendido proporciona soporte para los siguientes gestores de transacciones externos:

Gestores de transacciones compatibles con XA
El cliente transaccional extendido proporciona la interfaz de gestor de recursos XA para dar soporte a gestores de transacciones compatibles con XA como por ejemplo, CICS y Tuxedo.

Windows **Microsoft Transaction Server (sólo sistemas Windows)**
Sólo en sistemas Windows, la interfaz del gestor de recursos XA también da soporte a Microsoft Transaction Server (MTS). El soporte MTS de IBM MQ suministrado con el cliente transaccional extendido proporciona el puente entre MTS y la interfaz del gestor de recursos XA.

WebSphere Application Server
WebSphere Application Server 6 y posteriores incluyen un proveedor de mensajería de IBM MQ , por lo que no es necesario utilizar el cliente transaccional extendido.

La configuración de los gestores de transacciones compatibles con XA

Primero configure el cliente base de IBM MQ y luego configure la función transaccional extendida, utilizando la información contenida en estos temas.

Nota: En esta sección se presupone que tiene conocimientos básicos de la interfaz XA como publica The Open Group en *Distributed Transaction Processing: The XA Specification*.

Para configurar un cliente transaccional extendido, primero debe configurar el cliente base de IBM MQ como se describe en:

- AIX Instalación de un cliente IBM MQ en AIX
- Linux Instalación de un cliente IBM MQ en Linux

- Windows Instalación de un cliente IBM MQ en Windows
- IBM i Instalación de un cliente IBM MQ en IBM i

Utilizando la información para la plataforma, puede configurar la función transaccional extendida para un gestor de transacciones compatible con XA como CICS y Tuxedo.

Un gestor de transacciones se comunica con un gestor de colas como un gestor de recursos utilizando el mismo canal MQI que ha utilizado la aplicación cliente que está conectada al gestor de colas. Cuando el gestor de transacciones emite una llamada de función (xa_) de gestor de recursos, el canal MQI se utiliza para reenviar la llamada al gestor de colas y para recibir la salida del gestor de colas.

El gestor de transacciones puede iniciar el canal MQI emitiendo una llamada xa_open para abrir el gestor de colas como un gestor de recursos, o bien la aplicación cliente puede iniciar el canal MQI emitiendo una llamada MQCONN o MQCONNX.

- Si el gestor de transacciones inicia el canal MQI y, posteriormente, la aplicación cliente llama a MQCONN o MQCONNX en la misma hebra, la llamada a MQCONN o MQCONNX se completa satisfactoriamente y se devuelve un manejador de conexión a la aplicación. La aplicación no recibe un código de terminación MQCC_WARNING con un código de razón MQRC_ALREADY_CONNECTED.
- Si la aplicación cliente inicia el canal MQI y, posteriormente, el gestor de transacciones llama a xa_open en la misma hebra, la llamada a xa_open se reenvía al gestor de colas utilizando el canal MQI.

En una situación de recuperación después de una anomalía, cuando no se estén ejecutando aplicaciones cliente, el gestor de transacciones puede utilizar un canal MQI destinado a recuperar cualquier unidad de trabajo incompleta en la cual el gestor de colas haya participado en el momento de la anomalía.

Tenga en cuenta las condiciones siguientes al utilizar un cliente transaccional extendido con un gestor de transacciones compatible con XA:

- En una sola hebra, una aplicación cliente sólo puede conectarse a un gestor de colas a la vez. Esta restricción sólo se aplica cuando se utiliza un cliente transaccional extendido; una aplicación cliente que utilice un cliente base IBM MQ puede conectarse a más de un gestor de colas simultáneamente en una sola hebra.
- Cada hebra de una aplicación cliente puede conectarse a un gestor de colas diferente.
- Una aplicación cliente no puede utilizar manejadores de conexión compartidos.

Para configurar la función transaccional extendida, deberá proporcionar la siguiente información al gestor de transacciones para cada gestor de colas que actúe como gestor de recursos:

- Una serie de caracteres xa_open
- Un puntero a una estructura de conmutación XA

Cuando el gestor de transacciones llama a xa_open para abrir el gestor de colas como gestor de recursos, pasa la serie de caracteres xa_open al cliente transaccional extendido como argumento, *xa_info*, en la llamada. El cliente transaccional extendido utiliza la información de la serie de caracteres xa_open de la siguiente manera:

- Para iniciar un canal MQI en el gestor de colas del servidor, si la aplicación cliente todavía no se ha iniciado.
- Para comprobar que el gestor de colas que el gestor de transacciones abre como un gestor de recursos es el mismo que el gestor de colas al cual la aplicación cliente se conecta.
- Para ubicar las funciones ax_reg y ax_unreg del gestor de transacciones si el gestor de colas utiliza el registro dinámico.

Para conocer el formato de una serie de caracteres xa_open y para obtener más detalles sobre cómo utiliza el cliente transaccional extendido la información de la serie de caracteres xa_open, consulte “El formato de una serie xa_open” en la página 23.

Una estructura de conmutación XA permite al gestor de transacciones localizar las funciones xa_ proporcionadas por el cliente transaccional extendido y especifica si el gestor de colas utiliza el registro

dinámico. Para obtener información sobre las estructuras de conmutación XA proporcionadas por un cliente transaccional extendido, consulte “Estructuras de conmutación XA” en la página 27.

Para obtener información sobre cómo configurar la función transaccional extendida para un gestor de transacciones determinado y para cualquier otra información sobre la utilización del gestor de transacciones con un cliente transaccional extendido, consulte las secciones siguientes:

- “Configuración de un cliente transaccional extendido para CICS” en la página 28
- “Configuración de un cliente transaccional extendido para Tuxedo” en la página 30

Conceptos relacionados

“Los parámetros CHANNEL, TRPTYPE, CONNAME y QMNAME de la serie de caracteres xa_open” en la página 25
Utilice esta información para comprender cómo el cliente transaccional extendido utiliza estos parámetros para determinar el gestor de colas al que debe conectarse.

“Proceso de errores adicionales para xa_open” en la página 26
La llamada xa_open no se realiza satisfactoriamente en determinadas circunstancias.

Tareas relacionadas

“Utilización del cliente transaccional extendido con canales TLS” en la página 28
No puede configurar un canal TLS utilizando la serie xa_open. Siga estas instrucciones para utilizar la tabla de definiciones de canal de cliente (CCDT).

Referencia relacionada

“Los parámetros TPM y AXLIB” en la página 26
Un cliente transaccional extendido utiliza los parámetros TPM y AXLIB para localizar las funciones ax_reg y ax_unreg del gestor de transacciones. Estas funciones sólo se utilizan si el gestor de colas utiliza un registro dinámico.

“Recuperación después de una anomalía en el proceso transaccional extendido” en la página 26
Después de una anomalía, el gestor de transacciones debe poder recuperar cualquier unidad de trabajo incompleta. Para ello, el gestor de transacciones debe poder abrir como gestor de recursos cualquier gestor de colas que participe en una unidad de trabajo incompleta en el momento en que se produce la anomalía.

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

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

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

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

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

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

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

Notes:

- The change is not enabled by default. To make use of the change you must specify the keyword CSQSERVICE1 (in upper case) anywhere in the description field of the SVRCONN channel used by the XA client.
- Channels with the CSQSERVICE1 keyword have the following restrictions:

– GROUP unit of recovery disposition is not permitted. Only QMGR unit of recovery disposition is allowed. The disposition is determined by the name given on the xa_open call. If the queue sharing group name is used, then the XA connection requests a group unit of recovery.

  An xa_open call specifying the queue sharing group name in the **xa_info** parameter fails with *xaer_inval*.
– The *MQGMO_LOCK* and *MQGMO_UNLOCK* options are not permitted. An MQGET call with *MQGMO_LOCK* or *MQGMO_UNLOCK* fails with MQRC_ENVIRONMENT_ERROR.

The change was enabled at IBM MQ for z/OS 9.0 through APAR P173410

Related concepts
"La configuración de los gestores de transacciones compatibles con XA" on page 20
Primero configure el cliente base de IBM MQ y luego configure la función transaccional extendida, utilizando la información contenida en estos temas.

El formato de una serie xa_open

Una serie xa_open que contiene pares de nombres de parámetro y valores definidos.

Una serie xa_open tiene el formato siguiente:

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

donde *parm_name* es el nombre de un parámetro y *parm_value* es el valor de un parámetro. Los nombres de los parámetros no distinguen entre mayúsculas y minúsculas; a menos que se indique lo contrario, los valores de los parámetros distinguen entre mayúsculas y minúsculas. Puede especificar los parámetros en cualquier orden.

Los nombres, significados y valores válidos de los parámetros son los siguientes:

Nombre
Significado y valores válidos

CHANNEL
El nombre de un canal MQI.

Este es un parámetro opcional. Si se suministra este parámetro, también debe suministrarse el parámetro CONNAME.

TRPTYPE
El protocolo de comunicaciones para el canal MQI. Los protocolos siguientes son valores válidos:

LU62
SNA LU 6.2

NETBIOS
NetBIOS

SPX
IPX/SPX

TCP
TCP/IP

Este es un parámetro opcional. Si se omite, se adopta el valor predeterminado de TCP. Los valores del parámetro no distinguen entre mayúsculas y minúsculas.

CONNAME
La dirección de red del gestor de colas en el extremo del servidor del canal MQI. Los valores válidos de este parámetro dependen del valor del parámetro TRPTYPE:

LU62
Un nombre de destino simbólico que identifica una entrada de información complementaria CPI-C.

El nombre calificado para la red de una LU asociada no es un valor válido, ni es un alias de LU asociada. Esto se debe a que no hay ningún parámetro adicional para especificar un nombre de programa de transacción (TP) y un nombre de modalidad.

NETBIOS
Nombre de NetBIOS.

SPX
Una dirección de red de 4 bytes, una dirección de nodo de 6 bytes y un número de socket de 2 bytes opcional. Estos valores deben especificarse en notación hexadecimal. Un periodo debe separar la red y direcciones de nodo y el número de socket, si se ha suministrado, debe estar entre paréntesis. Por ejemplo:

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

Si se omite el número de socket, se adopta el valor predeterminado de 5e86.

TCP
Un nombre de host o una dirección IP, seguido opcionalmente de un número de puerto entre paréntesis. Si se omite el número de puerto, se adopta el valor predeterminado de 1414. Puede especificar varios hosts y puertos para un gestor de colas separándolos con punto y coma, por ejemplo:

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

Este es un parámetro opcional. Si se suministra este parámetro, también debe suministrarse el parámetro CHANNEL

QMNAME
El nombre del gestor de colas al final del servidor del canal MQI. El nombre puede estar o un asterisco único (*) ni el nombre se puede iniciar con un asterisco. Esto significa que el parámetro debe identificar un gestor de colas específico por el nombre.

Este parámetro es obligatorio.

Cuando una aplicación cliente está conectada a un gestor de colas específico, cualquier recuperación de transacción debe procesarse mediante el mismo gestor de colas.

z/OS Si la aplicación se conecta a un gestor de colas z/OS, la aplicación puede especificar el nombre de un gestor de colas específico o el nombre de un grupo de compartición de colas (QSG). Mediante el uso del nombre del gestor de colas o el nombre del grupo de compartición de colas, la aplicación controla si participa en una transacción con una unidad QMGR de disposición de recuperación o una unidad GROUP de disposición de recuperación. La unidad GROUP de disposición de recuperación habilita la recuperación de la transacción para ser procesada en cualquier miembro del QSG. Para utilizar unidades GROUP de recuperación, el atributo del gestor de colas **GROUPUR** debe estar habilitado. Para obtener más información sobre la utilización de la unidad de recuperación GROUP, consulte Disposición de unidad de recuperación en un grupo de compartición de colas.

TPM
El gestor de transacción que se está utilizando. Los valores válidos son CICS y TUXEDO.

Un cliente transaccional extendido utiliza ese parámetro y el parámetro AXLIB con la misma finalidad. Para obtener más información sobre estos parámetros, consulte TPM y parámetros AXLIB.

Este es un parámetro opcional. Los valores del parámetro no distinguen entre mayúsculas y minúsculas.

AXLIB
El nombre de la biblioteca que contiene las funciones ax_reg y ax_unreg del gestor de transacciones.

Este es un parámetro opcional.

UID
El ID de usuario que se proporciona al gestor de colas para autenticación. Si se suministra este parámetro, también debe suministrarse el parámetro **PWD**. Si se suministran y autentican el ID de usuario y la contraseña, se utiliza el ID de usuario para identificar la conexión del gestor de transacciones. El ID de usuario y la contraseña rellenan el objeto MQCSP en la llamada MQCONNX.

Los parámetros **UID** y **PWD** son válidos para los enlaces de cliente y servidor.

PWD
La contraseña que se proporciona al gestor de colas para autenticación. Si se suministra este parámetro, también debe suministrarse el parámetro **UID**.

Aviso: En algunos casos, la contraseña en la estructura MQCSP para una aplicación cliente se enviará por una red en texto sin formato. Para asegurarse de que las contraseñas de la aplicación cliente están protegidas adecuadamente, consulte IBM MQProtección de contraseña de CSP.

Este es un ejemplo de serie xa_open:

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

Los parámetros CHANNEL, TRPTYPE, CONNAME y QMNAME de la serie de caracteres xa_open

Utilice esta información para comprender cómo el cliente transaccional extendido utiliza estos parámetros para determinar el gestor de colas al que debe conectarse.

Si los parámetros **CHANNEL** y **CONNAME** se suministran en la serie xa_open, el cliente transaccional ampliado utiliza estos parámetros y el parámetro **TRPTYPE** para iniciar un canal MQI en el gestor de colas del servidor.

Si los parámetros **CHANNEL** y **CONNAME** no se proporcionan en la serie xa_open, el cliente transaccional ampliado utiliza el valor de la variable de entorno MQSERVER para iniciar un canal MQI. Si la variable de entorno MQSERVER no está definida, el cliente transaccional ampliado utiliza la entrada en la definición de canal de cliente identificada por el parámetro **QMNAME**.

En cada uno de estos casos, el cliente transaccional ampliado comprueba que el valor del parámetro **QMNAME** es el nombre del gestor de colas en el extremo del servidor del canal MQI. En caso contrario, la llamada xa_open no se ejecutará correctamente y el gestor de transacciones reportará dicha anomalía a la aplicación.

z/OS Si la aplicación utiliza un nombre de grupo de compartición de colas en el campo del parámetro **QMNAME** y la propiedad GROUPUR está inhabilitada en el gestor de colas al que se conecta, la llamada xa_open falla.

z/OS Si el cliente de aplicaciones se conecta a un gestor de colas z/OS , puede especificar un nombre de grupo de compartición de colas (QSG) para el parámetro **QMNAME** . Esto permite al cliente de aplicación participar en una transacción con una unidad GROUP de disposición de recuperación. Para obtener más información sobre la disposición de la unidad de recuperación GROUP, consulte Disposición de la unidad de recuperación.

Cuando la aplicación cliente posteriormente llama a MQCONN o MQCONNX en la misma hebra que el gestor de transacciones utilizó para emitir la llamada xa_open, la aplicación recibe un manejador de conexión para el canal MQI que inició la llamada xa_open. No se ha iniciado un segundo canal MQI. El cliente transaccional extendido comprueba que el valor del parámetro **QMgrName** de la llamada MQCONN o MQCONNX sea el nombre del gestor de colas en el extremo del servidor del canal MQI. En caso contrario, la llamada MQCONN o MQCONNX no se ejecuta correctamente con un código de razón de MQRC_ANOTHER_Q_MGR_CONNECTED. Si el valor del parámetro **QMgrName** está en blanco, es un asterisco (*) o empieza por un asterisco, la llamada MQCONN o MQCONNX no se ejecuta correctamente con un código de error de MQRC_Q_MGR_NAME_ERROR.

Si la aplicación cliente ya ha iniciado un canal MQI llamando MQCONN o MQCONNX antes de que el gestor de transacciones llame a xa_open en la misma hebra, el gestor de transacciones utilizará este canal MQI.

No se ha iniciado un segundo canal MQI. El cliente transaccional ampliado comprueba que el valor del parámetro **QMNAME** en la serie xa_open es el nombre del gestor de colas de servidor. En caso contrario, la llamada xa_open no se ejecutará correctamente.

Si una aplicación cliente inicia primero un canal MQI, el valor del parámetro **QMgrName** de la llamada MQCONN o MQCONNX podrá estar en blanco, ser un asterisco (*) o empezar por un asterisco. No obstante, en estas circunstancias deberá asegurarse de que el gestor de colas al que se conecta la aplicación sea el mismo que el gestor de colas que el gestor de transacciones va a abrir como un gestor de recursos cuando posteriormente llame a xa_open en la misma hebra. Por lo tanto, puede que tenga algunos problemas si el valor del parámetro *QMgrName* identifica el gestor de colas explícitamente por el nombre.

Los parámetros TPM y AXLIB

Un cliente transaccional extendido utiliza los parámetros TPM y AXLIB para localizar las funciones ax_reg y ax_unreg del gestor de transacciones. Estas funciones sólo se utilizan si el gestor de colas utiliza un registro dinámico.

Si el parámetro TPM se proporciona en una serie de caracteres xa_open, pero no se proporciona el parámetro AXLIB, el cliente transaccional extendido supone un valor para el parámetro AXLIB basándose en el valor del parámetro TPM. Consulte la Tabla 3 en la página 26 para obtener los valores del parámetro AXLIB asumidos.

Tabla 3. Valores asumidos del parámetro AXLIB

Valor de TPM	Plataforma	Valor asumido de AXLIB
CICS	AIX	/usr/lpp/encina/lib/libEncServer.a(EncServer_shr.o)
CICS	Sistemas Windows	libEncServer
Tuxedo	AIX	/usr/lpp/tuxedo/lib/libtux.a(libtux.so.60)
Tuxedo	Sistemas Windows	libtux

Si se proporciona el parámetro AXLIB en una serie de caracteres xa_open, el cliente transaccional extendido utiliza el valor para alterar temporalmente cualquier valor supuesto basándose en el valor del parámetro TPM. El parámetro AXLIB también puede utilizarse para un gestor de transacciones en el que el parámetro TPM no tiene un valor específico.

Proceso de errores adicionales para xa_open

La llamada xa_open no se realiza satisfactoriamente en determinadas circunstancias.

Los temas de este apartado describen situaciones en las que la llamada xa_open no se ejecuta satisfactoriamente. También es anómala si se produce alguna de las situaciones siguientes:

- Hay errores en la serie xa_open.
- No hay suficiente información para iniciar un canal MQI.
- Se registra un problema mientras se intenta iniciar un canal MQI (el gestor de colas del servidor no está en ejecución, por ejemplo).

Recuperación después de una anomalía en el proceso transaccional extendido

Después de una anomalía, el gestor de transacciones debe poder recuperar cualquier unidad de trabajo incompleta. Para ello, el gestor de transacciones debe poder abrir como gestor de recursos cualquier gestor de colas que participe en una unidad de trabajo incompleta en el momento en que se produce la anomalía.

Por lo tanto, debe asegurarse de que todas las unidades de trabajo incompletas se hayan resuelto antes de realizar los cambios.

De forma alternativa, debe asegurarse de que los cambios de configuración no afecten a la posibilidad del gestor de transacciones de abrir los gestores de colas necesarios. Estos son ejemplos de dichos cambios de configuración:

- Cambio del contenido de una serie de caracteres xa_open
- Cambio del valor de la variable de entorno MQSERVER
- Cambio de las entradas de la tabla de definiciones de canal de cliente (CCDT)
- Supresión de una definición de canal de conexión con el servidor

Estructuras de conmutación XA

Se proporcionan dos estructuras de conmutación XA con el cliente transaccional extendido en cada plataforma.

Estas estructuras de conmutación son:

MQRMIXASwitch
Un gestor de transacciones utiliza esta estructura de conmutación cuando un gestor de colas, que actúe como gestor de recursos, no utiliza el registro dinámico.

MQRMIXASwitchDynamic
Un gestor de transacciones utiliza esta estructura de conmutación cuando un gestor de colas, que actúe como gestor de recursos, utiliza el registro dinámico.

Estas estructuras de conmutación se ubican en las bibliotecas que se indican en la Tabla 4 en la página 27.

Tabla 4. Bibliotecas de IBM MQ que contienen las estructuras de conmutación XA

Plataforma	**Biblioteca que contiene las estructuras de conmutación XA**
AIX AIX Linux Linux	*MQ_INSTALLATION_PATH*/lib/libmqcxa
Sistemas Windows Windows	*MQ_INSTALLATION_PATH*\bin\mqcxa.dll [1]

MQ_INSTALLATION_PATH representa el directorio de alto nivel en el que está instalado IBM MQ.

El nombre del gestor de recursos de IBM MQ en cada estructura de conmutación es MQSeries_XA_RMI, pero muchos gestores de colas pueden compartir la misma estructura de conmutación.

Conceptos relacionados

“Registro dinámico y proceso transaccional extendido” en la página 27
El uso del registro dinámico es una forma de optimización ya que puede reducir el número de llamadas a función xa_ emitidas por el gestor de transacciones.

Registro dinámico y proceso transaccional extendido
El uso del registro dinámico es una forma de optimización ya que puede reducir el número de llamadas a función xa_ emitidas por el gestor de transacciones.

Si un gestor de colas no utiliza el registro dinámico, el gestor de transacciones implicará al gestor de colas en cada unidad de trabajo. El gestor de transacciones realiza esto mediante una llamada a xa_start, xa_end y xa_prepare, aunque el gestor de colas no tenga ningún recurso que se actualice en la unidad de trabajo.

Si un gestor de colas utiliza el registro dinámico, un gestor de transacciones se inicia con la presunción de que el gestor de colas no participa en una unidad de trabajo y no llama a xa_start. El gestor de colas participará en la unidad de trabajo sólo si sus recursos se actualizan en el control de punto de sincronismo. Si ocurre esto, el cliente transaccional extendido llama a ax_reg para registrar la participación del gestor de colas.

Utilización del cliente transaccional extendido con canales TLS

No puede configurar un canal TLS utilizando la serie xa_open. Siga estas instrucciones para utilizar la tabla de definiciones de canal de cliente (CCDT).

Acerca de esta tarea

Debido al tamaño limitado de la serie xa_open xa_info, no es posible pasar toda la información necesaria para configurar un canal TLS utilizando el método de serie xa_open de conexión a un gestor de colas. Por consiguiente, debe utilizar la tabla de definiciones de canal de cliente o, si el gestor de transacciones lo permite, crear el canal con MQCONNX antes de emitir la llamada xa_open.

Para utilizar la tabla de definiciones de canal de cliente, siga estos pasos:

Procedimiento

1. Especifique una serie xa_open que sólo contenga el parámetro obligatorio qmname (nombre de gestor de colas), por ejemplo: `XA_Open_String=qmname=MYQM`
2. Utilice un gestor de colas para definir un canal CLNTCONN (conexión de cliente) con los parámetros TLS necesarios. Incluya el nombre de gestor de colas en el atributo QMNAME de la definición CLNTCONN. Éste se comparará con el qmname de la serie xa_open.
3. Ponga la definición CLNTCONN disponible para el sistema cliente en una tabla de definiciones de canal de cliente (CCDT) o, en Windows, en Active Directory.
4. Si está utilizando una CCDT, identifique la CCDT que contiene la definición del canal CLNTCONN utilizando las variables de entorno MQCHLLIB y MQCHLTAB. Establezca estas variables en los entornos utilizados por la aplicación cliente y el gestor de transacciones.

Resultados

Esto proporciona al gestor de transacciones una definición de canal para el gestor de colas apropiado con los atributos TLS necesarios para autenticar correctamente, incluido SSLCIPH, la CipherSpec.

Configuración de un cliente transaccional extendido para CICS

Configure un cliente transaccional extendido para su uso en CICS añadiendo una definición de recurso XAD a una región CICS.

Añada la definición de recurso XA mediante el mandato de definición de recursos en línea (RDO) CICS, **`cicsadd`**. En la definición de recurso XAD se especifica la siguiente información:

- Una serie de caracteres xa_open
- El nombre completo de la vía de acceso de un archivo de carga de conmutación

Se proporciona un archivo de carga conmutada para que sea utilizado por CICS en cada una de las plataformas siguientes:

- 

AIX
- 

Windows

Cada archivo de carga de conmutación contiene una función que devuelve un puntero a la estructura de conmutación XA que se utiliza para el registro dinámico, MQRMIXASwitchDynamic. Consulte la Tabla 5 en la página 29 para obtener el nombre de la vía de acceso de cada archivo de carga de conmutación.

Tabla 5. Los archivos de carga de conmutación	
Plataforma	**Archivo de carga de conmutación**
AIX Linux	*MQ_INSTALLATION_PATH*/lib/mqczsc
Windows	*MQ_INSTALLATION_PATH*\bin\mqcc4swi.dll [1]

MQ_INSTALLATION_PATH representa el directorio de alto nivel en el que está instalado IBM MQ.

El siguiente es un ejemplo de una definición de recurso XAD para sistemas Windows:

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

Para obtener más información sobre cómo añadir una definición de recurso XAD a una región CICS, consulte las publicaciones *CICS Administration Reference* y *CICS Administration Guide* correspondientes a su plataforma.

Tenga en cuenta la información siguiente sobre la utilización de CICS con un cliente transaccional extendido:

- Sólo se puede añadir una definición de recurso XAD para IBM MQ a una región CICS. Esto significa que sólo un gestor de colas puede estar asociado con una región y que todas las aplicaciones CICS que se ejecutan en la región sólo pueden conectarse a dicho gestor de colas. Si desea ejecutar aplicaciones CICS que se conectan a un gestor de colas diferente, debe ejecutar las aplicaciones en otra región.
- Cada servidor de aplicaciones de una región llama a xa_open mientras se está inicializando e inicia un canal MQI en el gestor de colas asociado con la región. Esto significa que el gestor de colas debe iniciarse antes de que se inicie un servidor de aplicaciones, de lo contrario la llamada xa_open no se ejecutará correctamente. Todas las aplicaciones de IBM MQ MQI client procesadas posteriormente por el servidor de aplicaciones utilizan el mismo canal MQI.
- Cuando se inicia un canal MQI y no hay ninguna salida de seguridad en el extremo del cliente del canal, el ID de usuario que fluye del sistema cliente al MCA de conexión con el servidor es `cics`. En algunos casos, el gestor de colas utiliza este ID de usuario para realizar comprobaciones de autorización, cuando el MCA de conexión con el servidor intenta acceder posteriormente a los recursos del gestor de colas en nombre de una aplicación cliente. Si se utiliza este ID de usuario para realizar las comprobaciones de autorización, deberá asegurarse de que tenga autorización para acceder a todos los recursos que necesite.

  Para obtener información sobre cuándo el gestor de colas utiliza este ID de usuario para comprobaciones de autorización, consulte Seguridad.
- Las salidas de terminación de tarea CICS que se proporcionan para su uso en sistemas cliente IBM MQ se listan en la Tabla 6 en la página 30. Estas salidas se configuran del mismo modo que se configuran las salidas correspondientes para los sistemas servidor de IBM MQ. Por consiguiente, para esta información, consulte Habilitación de salidas de usuario CICS.

Tabla 6. Salidas de terminación de tarea CICS		
Plataforma	**Origen**	**Biblioteca**
AIX AIX Linux Linux	amqzscgx.c	amqczscg
Sistemas Windows Windows	amqzscgn.c	mqcc1415.dll

Configuración de un cliente transaccional extendido para Tuxedo

Para configurar la definición de recurso XAD para que la utilice Tuxedo, actualice el archivo UBBCONFIG, y la tabla del gestor de recursos.

Para configurar la definición de recurso XAD para que la utilice Tuxedo, efectúe las acciones siguientes:

- En la sección GROUPS del archivo Tuxedo UBBCONFIG para una aplicación, utilice el parámetro **OPENINFO** para espedcificar una serie xa_open. Para obtener un ejemplo sobre cómo hacerlo, consulte el archivo de ejemplo UBBCONFIG, el cual se proporciona con los programas de ejemplo Tuxedo.

  AIX En las plataformas siguientes, el nombre del archivo es `ubbstxcx.cfg`:

  – AIX

  Windows Windows, el nombre del archivo es `ubbstxcn.cfg`.
- En la entrada para un gestor de colas en la tabla del gestor de recursos Tuxedo, especifique el nombre de una estructura de conmutación XA y el nombre de vía de acceso completo de la biblioteca que contiene la estructura:

  – AIX En AIX, especifique `udataobj/RM`.

  – Windows En Windows, especifique `udataobj\rm`.

  Para obtener un ejemplo de cómo realizar esta operación en cada plataforma, consulte Ejemplos de TUXEDO. Tuxedo da soporte al registro dinámico de un gestor de recursos, por lo tanto, puede utilizar tanto MQRMIXASwitch como MQRMIXASwitchDynamic.

Windows Servidor de transacciones de Microsoft

No es necesaria ninguna configuración adicional antes de poder utilizar Microsoft Transaction Server (MTS) como gestor de transacciones. No obstante, hay algunos puntos que deben tenerse en cuenta.

Tenga en cuenta la siguiente información sobre el uso de MTS con el cliente transaccional extendido:

- Una aplicación MTS siempre inicia un canal MQI cuando se conecta a un gestor de colas del servidor. MTS, en su función de gestor de transacciones, utiliza el mismo canal MQI para comunicarse con el gestor de colas.
- Después de una anomalía, MTS debe poder recuperar cualquier unidad de trabajo incompleta. Para ello, MTS debe poder comunicarse con cualquier gestor de colas que haya participado en una unidad de trabajo incompleta en el momento de la anomalía.

  Cuando una aplicación MTS se conecta a un gestor de colas de servidor e inicia un canal MQI, el cliente transaccional extendido extrae suficiente información de los parámetros de la llamada MQCONN o MQCONNX para permitir que el canal se reinicie después de la anomalía, si es necesario. El cliente transaccional extendido pasa la información a MTS y MTS registra la información en las anotaciones cronológicas.

  Si la aplicación MTS emite una llamada MQCONN, esta información será simplemente el nombre del gestor de colas. Si la aplicación MTS emite una llamada MQCONNX y proporciona una estructura de

definición de canal, MQCD, en la información también se incluirá el nombre del canal MQI, la dirección de red del gestor de colas del servidor y el protocolo de comunicación del canal.

En una situación de recuperación, MTS devuelve esta información al cliente transaccional extendido y éste la utiliza para reiniciar el canal MQI.

Si en algún momento necesita cambiar la información de configuración, asegúrese de que todas las unidades de trabajo incompletas se han resuelto antes de realizar los cambios. De forma alternativa, asegúrese de que los cambios de configuración no afecten la posibilidad del cliente transaccional extendido de reiniciar un canal MQI utilizando la información registrada por MTS. Estos son ejemplos de dichos cambios de configuración:

- Cambio del valor de la variable de entorno MQSERVER
- Cambio de las entradas de la tabla de definiciones de canal de cliente (CCDT)
- Supresión de una definición de canal de conexión con el servidor

- Tenga en cuenta las siguientes condiciones cuando utilice un cliente transaccional extendido con MTS:
  - En una sola hebra, una aplicación cliente sólo puede conectarse a un gestor de colas a la vez.
  - Cada hebra de una aplicación cliente puede conectarse a un gestor de colas diferente.
  - Una aplicación cliente no puede utilizar manejadores de conexión compartidos.

Definición de canales MQI

Para crear un nuevo canal, tiene que crear **dos** definiciones de canal, una para cada extremo de la conexión, utilizando el mismo nombre de canal y tipos de canal compatibles. En este caso, los tipos de canal son *server-connection* y *client-connection*.

Canales definidos por el usuario

Cuando el servidor no define automáticamente los canales, hay dos maneras de crear las definiciones de canal y de otorgar a la aplicación IBM MQ en la máquina de IBM MQ MQI client acceso al canal.

Estos dos métodos se describen de forma exhaustiva:

1. Cree una definición de canal en el cliente IBM MQ y otra en el servidor.

   Esto se aplica a cualquier combinación de plataformas de servidor y IBM MQ MQI client. Utilícela cuando empiece en el sistema o para probar la instalación.

   Consulte “Creación de definiciones de conexión de servidor y de conexión de cliente en plataformas diferentes” en la página 37 para obtener información detallada sobre cómo utilizar este método.
2. Cree ambas definiciones de canal en la máquina de servidor.

   Utilice este método cuando esté configurando varios canales y máquinas de IBM MQ MQI client al mismo tiempo.

   Consulte “Creación de definiciones de conexión de servidor y de conexión de cliente en el servidor” en la página 43 para obtener información detallada sobre cómo utilizar este método.

Canales definidos automáticamente

Multi

Los productos de IBM MQ en Multiplatforms incluyen una característica que puede crear automáticamente una definición de canal en el servidor si no existe.

Si se recibe una solicitud de conexión de entrada de un cliente y no se encuentra una definición de conexión de servidor adecuada en ese gestor de colas, IBM MQ crea una definición automáticamente y la añade al gestor de colas. La definición automática se basa en la definición del canal de conexión de servidor predeterminado SYSTEM.AUTO.SVRCONN. Debe habilitar la definición automática de las definiciones de conexión de servidor actualizando el objeto de gestor de colas mediante el mandato

ALTER QMGR con el parámetro CHAD (o el mandato PCF Change Queue Manager con el parámetro ChannelAutoDef).

Conceptos relacionados

“Función de control de canales” en la página 240
La función de control de canales proporciona recursos para definir, supervisar y controlar canales.

ALW Creación y utilización de canales AMQP

Cuando instala el soporte de IBM MQ para el componente de servicio AMQP en la instalación de IBM MQ, puede ejecutar mandatos MQSC de IBM MQ (**runmqsc**) para definir, alterar, suprimir, iniciar y detener un canal. También puede ver el estado de un canal.

Antes de empezar

Esta tarea presupone que ha instalado el canal AMQP. Para ello, ha seleccionado el componente de servicio AMQP al instalar IBM MQ. Para obtener más información, siga el enlace de la plataforma y, después, busque la fila de la tabla para "Servicio AMQP":

- AIX Componentes de IBM MQ para sistemas AIX
- Linux Componentes de IBM MQ para sistemas Linux
- Linux Componentes de IBM MQ Debian para sistemas Linux Ubuntu
- Windows Características de IBM MQ para sistemas Windows

Nota: Consulte Reinicio del servicio IBM MQ para AMQP para obtener un ejemplo de un componente SERVICE y más información si el servicio AMQP deja de funcionar correctamente.

Esta tarea también presupone que tiene un gestor de colas existente.

Para realizar una conexión de prueba con el gestor de colas, puede utilizar cualquier cliente AMQP que implemente el protocolo OASIS AMQP 1.0, como por ejemplo clientes MQ Light y Apache Qpid como Apache Qpid Proton y Apache Qpid JMS.

A partir de IBM MQ 9.3.0, solo puede utilizar el canal predeterminado, SYSTEM.DEF.AMQP, para probar las conexiones de MQ Light con el gestor de colas. El procedimiento siguiente utiliza el canal predeterminado.

Esta tarea se basa en el cliente Node.js de MQ Light. Sin embargo, los pasos relacionados con el gestor de colas IBM MQ son los mismos para cualquier cliente.

Nota: Los canales AMQP no dan soporte a los servicios AMQP definidos por usuario. Los canales AMQP solo dan soporte al servicio predeterminado del sistema SYSTEM.AMQP.SERVICE. Sólo puede definir una instancia de este servicio por gestor de colas.

Procedimiento

1. Inicie **runmqsc** desde el directorio *mqinstall*/bin/:

   ```
   runmqsc QMNAME
   ```

2. (Solo es necesario si el gestor de colas es V 9.4.0 V 9.4.0 de IBM MQ 9.4.0 o si es IBM MQ 9.0.4 o anterior.) Compruebe que la función AMQP está instalada y funciona correctamente.
   Utilice el mandato **START SERVICE** para iniciar el servicio IBM MQ , que controla la JVM:

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

Nota: V 9.4.0 V 9.4.0 Desde IBM MQ 9.4.0 el SYSTEM.AMQP.SERVICE tiene su atributo **CONTROL** establecido en MANUAL. Esto impide que el servicio se inicie cuando se inicia el gestor de colas. El establecimiento de la propiedad **CONTROL** en QMGRse inicia automáticamente cuando se inicia el gestor de colas.

De IBM MQ 9.1 a IBM MQ 9.3, el SYSTEM.AMQP.SERVICE tiene su atributo **CONTROL** establecido en QMGR.

3. Establezca el ID de usuario MCAUSER.

   Cuando un cliente AMQP se conecta a un canal, el canal especifica un ID de usuario MCAUSER, que se utiliza en las conexiones con el gestor de colas. El valor predeterminado es MCAUSER es dejarlo en blanco. Antes de que los clientes AMQP puedan conectarse al gestor de colas, debe especificar un valor MCAUSER , que debe ser un usuario válido de IBM MQ que esté autorizado para publicar y suscribirse en los temas de IBM MQ .

   a) Utilice el mandato **ALTER CHANNEL** para establecer el ID de usuario MCAUSER :

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

   b) Utilice los dos mandatos **setmqaut** siguientes para autorizar al ID de usuario MCAUSER a publicar y suscribirse a temas:

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

   y

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

   Si el canal se está ejecutando mientras se ha añadido o modificado el ID de usuario MCAUSER, debe detener y reiniciar el canal.

   Nota: Si el ID de usuario MCAUSER no está establecido, o el ID de usuario MCAUSER no está autorizado para publicar o suscribirse a temas IBM MQ, recibirá un mensaje de error en el cliente de AMQP.

4. Utilice el mandato **START CHANNEL** para iniciar el SYSTEM.DEF.AMQP :

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

5. Si desea comprobar el estado del canal, utilice el mandato **DISPLAY CHSTATUS** :

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

   Cuando el canal se ejecuta correctamente, STATUS(RUNNING) se visualiza en la salida del mandato.

6. Cambie el puerto predeterminado.

   El puerto predeterminado para conexiones AMQP 1.0 es 5672. Si ya está utilizando el puerto 5672, que es posible si previamente ha instalado MQ Light, debe cambiar el puerto que utiliza el canal AMQP. Utilice el mandato **ALTER CHANNEL** para cambiar el puerto:

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

7. Si no desea bloquear ni filtrar las conexiones al canal AMQP utilizando reglas de autenticación de canal (CHLAUTH), inhabilite la autenticación de canal en el gestor de reglas, del modo siguiente:

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

   No se recomienda inhabilitar la autenticación de conexión en un gestor de colas de producción. Solo debería inhabilitar la autenticación de conexión en un entorno de desarrollo.

De forma alternativa, configure las reglas de autenticación del canal de gestor de colas para permitir conexiones específicas al canal AMQP.

8. Opcional: Si desea habilitar el cifrado SSL/TLS en el canal, utilizando el repositorio de claves configurado para el gestor de colas, debe establecer el atributo SSLCIPH para el canal en una especificación de cifrado apropiada. De forma predeterminada, la especificación de cifrado está en blanco, lo que significa que el cifrado SSL/TLS no se utiliza en el canal. Utilice el mandato **ALTER CHANNEL** para establecer una especificación de cifrado. Por ejemplo:

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

Además, existen varias opciones de configuración de canal distintas asociadas al cifrado SSL/TLS que puede establecer del modo siguiente:

- De forma predeterminada, el certificado del repositorio de claves del gestor de colas con la etiqueta correspondiente al atributo **CERTLABL** del gestor de colas es el nombre utilizado por el cifrado SSL/TLS para el canal. Puede seleccionar un certificado distinto estableciendo **CERTLABL**. Utilice el mandato **ALTER CHANNEL** para especificar la etiqueta para el certificado necesario.

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

- Puede establecer el canal para que requiera un certificado de las conexiones de cliente SSL/TLS. Puede seleccionar si se necesita un certificado de una conexión de cliente SSL/TLS estableciendo el atributo **SSLCAUTH**. Utilice el mandato **ALTER CHANNEL** para establecer si se requiere un certificado de una conexión de cliente SSL/TLS. Por ejemplo:

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

- Si establece el atributo **SSLCAUTH** en REQUIRED, se puede comprobar el nombre distinguido (DN) del certificado del cliente. Para comprobar el nombre distinguido del certificado del cliente, establezca el atributo **SSLPEER**. Utilice el mandato **ALTER CHANNEL** para comprobar el nombre distinguido del certificado del cliente. Por ejemplo:

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

De forma alternativa, también puede utilizar registros de autenticación de canal para permitir o bloquear conexiones porque este método ofrece una mayor granularidad en comparación con el uso del atributo **SSLPEER**. Para obtener más información sobre cómo establecer **SSLPEER** y utilizar los registros de autenticación de canal como alternativa, consulte Igual SSL.

9. Instale el cliente Node.js de MQ Light ejecutando el mandato siguiente:

```
npm install mqlight
```

10. Vaya al directorio `node_modules/mqlight/samples` y ejecute la aplicación del receptor de ejemplo:

- Si está utilizando el número de puerto predeterminado, puede ejecutar la aplicación receptora de ejemplo:

```
node recv.js
```

- Si ha configurado el canal AMQP para utilizar un número de puerto diferente, puede ejecutar la aplicación receptora de ejemplo con un parámetro para especificar el nuevo número de puerto:

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

Una conexión satisfactoria al canal predeterminado muestra el mensaje siguiente:

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

La aplicación está ahora conectada al gestor de colas y está a la espera de recibir mensajes. Está suscrito al tema `public`.

Nota: El `client-id` se genera automáticamente, a menos que especifique uno utilizando el parámetro `-i`.

11. En una nueva ventana de mandatos, vaya al directorio `node_modules/mqlight/samples` y ejecute la aplicación del remitente de ejemplo ejecutando el mandato siguiente:

```
node send.js
```

En la ventana de mandatos para la aplicación receptora, se muestra el mensaje `Hello World`.

12. Utilice el ejemplo **AMQSSUB** IBM MQ para recibir un mensaje de ejemplo de MQ Light.

En Linux y Windows, el ejemplo se puede encontrar en las ubicaciones siguientes:

- Linux Directorio *mqinstall*`/samp/bin` en Linux.
- Windows Directorio *mqinstall*`/Tools\c\Samples\Bin` en Windows.

a) Ejecute el ejemplo ejecutando el mandato siguiente:

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

b) Envíe un mensaje a la aplicación IBM MQ volviendo a ejecutar el mandato siguiente:

```
node send.js
```

13. Utilice el mandato **DEFINE CHANNEL** para crear más canales AMQP:

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

Cuando se define un canal, se debe iniciar manualmente, utilizando el mandato **START CHANNEL**:

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

Para comprobar que el canal se está ejecutando correctamente, puede ejecutar la aplicación receptora de ejemplo, especificando el puerto del nuevo canal:

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

Qué hacer a continuación

Puede utilizar los mandatos siguientes para mostrar las conexiones de IBM MQ, detener el canal y suprimir el canal:

DISPLAY CONN(*) TYPE(*CONN*) WHERE (*CHANNEL EQ SYSTEM.DEF.AMQP*)
Muestra la conexión de IBM MQ que ha realizado el canal AMQP en el gestor de colas.

DISPLAY CHSTATUS(*) CHLTYPE(*AMQP*) CLIENTID(*) ALL
Muestra una lista de los clientes AMQP conectados al canal especificado.

STOP CHANNEL (*MY.AMQP.CHANNEL*)
Detiene un canal AMQP y cierra el puerto en el que está a la escucha.

DELETE CHANNEL (*MY.AMQP.CHANNEL*)
Suprime los canales que ha creado.

Nota: No suprima el canal predeterminado SYSTEM.DEF.AMQP.

Puede determinar si la prestación AMQP se va a instalar en la instalación de IBM MQ y si va a tener un gestor de colas asociado, utilizando **runmqsc** o PCF:

- Mediante **runmqsc**, visualice los atributos del gestor de colas y compruebe AMQPCAP(YES).
- Mediante PCF, utilice el mandato **MQCMD_INQUIRE_Q_MGR** y confirme el valor de MQIA_AMQP_CAPABILITY.

Tareas relacionadas

Desarrollo de aplicaciones cliente AMQP

Protección de clientes de AMQP

Referencia relacionada

strmqm

ALW Eliminación del canal AMQP de los gestores de colas

Puede eliminar el canal AMQP de los gestores de colas eliminando carpetas del directorio de instalación.

Procedimiento

1. Detenga el gestor de colas.
2. Elimine el soporte de IBM MQ para las API del componente de servicio AMQP:
   - AIX En AIX, ejecute el mandato siguiente:

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

   - Linux En Linux, elimine el RPM AMQP. Si ha vuelto a empaquetar el RPM antes de instalarlo, especifique el nombre del RPM reempaquetado.

     ```
     rpm -e MQSeriesAMQP
     ```

   - Windows En Windows, elimine la carpeta amqp de la instalación de IBM MQ. Asegúrese de que no se elimina ningún otro archivo ni capeta de la vía de instalación de IBM MQ.
3. Reinicie el gestor de colas.

Tareas relacionadas

Desarrollo de aplicaciones cliente AMQP

Protección de clientes de AMQP

ALW Archivos de registro de canal AMQP

Los archivos de registro para canales AMQP se almacenan en el mismo directorio de datos de IBM MQ que los archivos de registro de IBM MQ.

El directorio de datos predeterminado en Windows es C:\ProgramData\IBM\MQ.

El directorio de datos predeterminado en Linux es /var/mqm.

El canal AMQP escribe la información de registro en los archivos de registro siguientes, que se encuentra en el directorio de datos de IBM MQ:

- amqp.stdout, escrito en la carpeta qmgrs/*QM-name*.
- amqp.stderr, escrito en la carpeta qmgrs/*QM-name*.
- amqp_*.log , escrito en la carpeta qmgrs/*QM-name*/errors.
- V 9.4.0 V 9.4.0 amqp_*.json , escrito en la carpeta qmgrs/*QM-name*/errors.

Si un cliente MQ Light recibe un error de autenticación o autorización, el administrador puede encontrar información detallada sobre el motivo del error de seguridad en el archivo `amqp_0.log` , V 9.4.0 V 9.4.0 `amqp_0.json file`, y los archivos `MQ AMQERR*.log` .

Los archivos FDC se crean como archivos `AMQP*.FDC`, que se graban en la carpeta *data-directory*`/errors`.

Algunos archivos de configuración se graban en el directorio `qmgrs/`*QM-name*`/amqp`.

V 9.4.0 V 9.4.0 Los registros con formato JSON en AMQP son opcionales y deben habilitarse manualmente. Para ello, modifique los registros de AMQP, registros de errores y archivos de configuración.

Conceptos relacionados

Registros de errores en AIX, Linux, and Windows

Tareas relacionadas

Desarrollo de aplicaciones cliente AMQP

Protección de clientes de AMQP

Habilitación de registros con formato JSON para AMQP

Creación de definiciones de conexión de servidor y de conexión de cliente en plataformas diferentes

Puede crear cada definición de canal en el sistema al que se aplica. Sin embargo, hay restricciones respecto a cómo se pueden crear definiciones de canal en un sistema cliente.

Acerca de esta tarea

En todas las plataformas se pueden utilizar mandatos de script de IBM MQ (MQSC), mandatos de formato de mandato programable (PCF) o IBM MQ Explorer para definir un canal de conexión de servidor en la máquina servidora.

z/OS En z/OS también puede utilizar los paneles de Operación y Control.

IBM i En IBM i también puede utilizar la interfaz de panel.

Puesto que los mandatos MQSC no están disponibles en una máquina en la que se ha instalado IBM MQ sólo como IBM MQ MQI client, debe utilizar diferentes formas de definir un canal de conexión de cliente en la máquina cliente.

Al ejecutar **runmqsc**, se aplican las consideraciones siguientes:

- Puede especificar el parámetro **-c** y, opcionalmente, el parámetro **-u** para conectar **runmqsc** como cliente al gestor de colas que desea administrar.
- Si utiliza el parámetro **-u** para suministrar un ID de usuario, se le solicita una contraseña coincidente.
- Si ha configurado el registro `CONNAUTH AUTHINFO` con `CHCKLOCL(REQUIRED)` o `CHCKLOCL(REQDADM)`, debe utilizar el parámetro **-u**, de lo contrario, no podrá administrar su gestor de colas con **runmqsc**.

Procedimiento

- Para definir un canal de conexión de servidor en el servidor, consulte “Definición de un canal de conexión del servidor en el servidor” en la página 38.
- Para crear un canal de conexión de cliente en un IBM MQ MQI client utilizando la variable de entorno **MQSERVER** , consulte “Creación de un canal de conexión de cliente en IBM MQ MQI client utilizando MQSERVER” en la página 38.

- Para crear un canal de conexión de cliente en un IBM MQ MQI clientutilizando la estructura MQCNO en una llamada MQCONNX, consulte “Creación de un canal de conexión de cliente en IBM MQ MQI client utilizando MQCNO” en la página 43.

Definición de un canal de conexión del servidor en el servidor

Inicie MQSC si es necesario y, a continuación, defina el canal de conexión del servidor.

Procedimiento

1. Opcional: Si está utilizando un servidor Multiplatforms, primero cree e inicie un gestor de colas y, a continuación, inicie los mandatos MQSC.
   a) Cree un gestor de colas, llamado QM1, por ejemplo:

   ```
   crtmqm QM1
   ```

   b) Inicie el gestor de colas:

   ```
   strmqm QM1
   ```

   c) Inicie los mandatos MQSC:

   ```
   runmqsc QM1
   ```

2. Defina un canal con el nombre que prefiera y el tipo de canal *conexión con el servidor*.

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

   Esta definición de canal se asocia al gestor de colas que se ejecuta en el servidor.
3. Utilice el mandato siguiente para otorgar a la entrada acceso de conexión al gestor de colas:

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

   - Donde SET CHLAUTH utiliza el nombre del canal definido en el paso anterior.
   - Donde *'dirección IP'* es la dirección IP del cliente.
   - Donde *'ID usuario'* es el ID que desea proporcionar al canal para el control de accesos a la colas de destino. Este campo es sensible a las mayúsculas y minúsculas.

   Puede elegir identificar la conexión de entrada mediante varios atributos distintos. En el ejemplo se utiliza la dirección IP. Entre los atributos alternativos se incluyen el ID de usuario del cliente y el nombre distinguido de asunto TLS. Para obtener más información, consulte Registros de autenticación de canal

Creación de un canal de conexión de cliente en IBM MQ MQI client utilizando MQSERVER

Puede definir un canal de conexión de cliente en una estación de trabajo cliente utilizando la variable de entorno **MQSERVER** .

Acerca de esta tarea

Puede utilizar la variable de entorno **MQSERVER** para especificar una definición simple de un canal de conexión de cliente. Se considera simple porque con este método puede especificar solamente algunos atributos del canal.

Si utiliza la variable de entorno **MQSERVER** para definir el canal entre la máquina IBM MQ MQI client y una máquina servidor, este es el único canal disponible para la aplicación y no se hace referencia a la tabla de definiciones de canal de cliente (CCDT).

Si la solicitud MQCONN o MQCONNX especifica un gestor de colas distinto al que está conectado el escucha, o si no se reconoce el parámetro **MQSERVER** *TransportType* , la solicitud MQCONN o MQCONNX falla con el código de retorno MQRC_Q_MGR_NAME_ERROR.

Linux AIX En AIX and Linux, puede definir **MQSERVER** como en uno de los ejemplos siguientes:

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

A continuación, todas las solicitudes MQCONN o MQCONNX intentan utilizar el canal que ha definido a menos que se haya hecho referencia a una estructura MQCD desde la estructura MQCNO proporcionada a MQCONNX, en cuyo caso el canal especificado por la estructura MQCD tiene prioridad sobre cualquier valor especificado por la variable de entorno **MQSERVER** .

La variable de entorno **MQSERVER** tiene prioridad sobre cualquier definición de canal de cliente a la que apunte las variables de entorno **MQCHLLIB** y **MQCHLTAB** .

Procedimiento

- En función de la plataforma, utilice uno de los mandatos siguientes para especificar la definición de canal con **MQSERVER**.
  - Windows En Windows, especifique una definición de canal simple como se indica a continuación:

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

    Por ejemplo:

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

  - Linux AIX En AIX and Linux, especifique una definición de canal simple como se indica a continuación:

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

    Por ejemplo:

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

  - IBM i En IBM i, especifique una definición de canal simple como se indica a continuación:

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

    Por ejemplo:

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

  Notas:

  - El *ChannelName* debe ser el mismo nombre que el definido en el servidor. No puede contener el carácter de barra inclinada (/) porque este carácter se utiliza para separar el nombre de canal, el

tipo de transporte y el nombre de conexión. Cuando se utiliza la variable de entorno **MQSERVER** para definir un canal de cliente, se utiliza una longitud máxima de mensaje (**MAXMSGL**) de 100 MB. Por consiguiente, el tamaño máximo de mensaje en vigor para el canal es el valor especificado en el canal SVRCONN en el servidor.

  - *TransportType* puede ser uno de LU62, TCP, NETBIOS, SPX, en función de la plataforma de cliente IBM MQ .
  - Linux AIX En AIX and Linux, *TransportType* distingue entre mayúsculas y minúsculas y debe estar en mayúsculas. Una llamada MQCONN o MQCONNX devuelve 2058 si no se reconoce el tipo de transporte
  - El *ConnectionName* es el nombre del servidor tal como se ha definido en el protocolo de comunicaciones (*TransportType*). Debe ser un nombre de red completo, por ejemplo, AMACHINE.ACOMPANY.COM(1414).
  - El *ConnectionName* puede ser una lista separada por comas de nombres de conexión. Los nombres de conexiones de la lista se utilizan de un modo similar para varias conexiones en una tabla de conexiones de cliente. La lista de nombres de conexión se puede utilizar como alternativa a los grupos de gestores de colas para especificar varias conexiones para que el cliente las intente. Si está configurando un gestor de colas de varias instancias, puede utilizar una lista de nombres de conexión para especificar distintas instancias de gestor de colas.

- Para cancelar **MQSERVER** y volver a la tabla de definiciones de canal de cliente a la que apuntan **MQCHLLIB** y **MQCHLTAB**, especifique el mandato siguiente:
  - Linux AIX En AIX and Linux:

```
unset MQSERVER
```

  - Windows En Windows:

```
SET MQSERVER=
```

Ejemplo

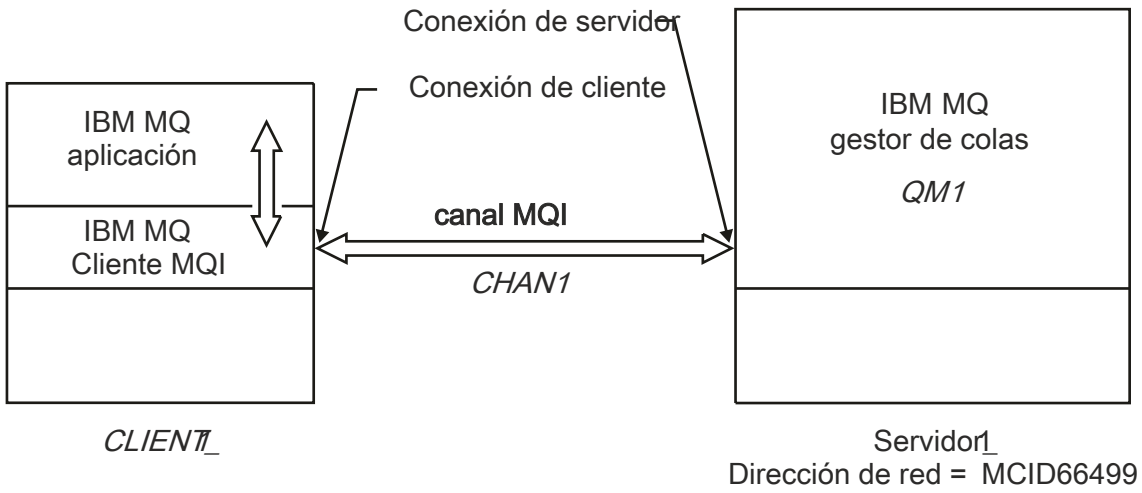


Figura 1. Ejemplo de una definición de canal simple

Para crear la definición de canal simple que se muestra en Figura 1 en la página 40, utilice los mandatos siguientes:

- Linux AIX En AIX and Linux:

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

- Windows En Windows:

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

Nota: Para obtener información sobre cómo cambiar el número de puerto TCP/IP, consulte “Cambio del puerto predeterminado TCP/IP” en la página 41.

Algunos ejemplos más de definiciones de canal simple son los siguientes:

- Windows En Windows:

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

- Linux AIX En AIX and Linux:

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

  donde B0X99 es el NombreConexión de LU 6.2.

- IBM i En IBM i:

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

En el IBM MQ MQI client, todas las solicitudes **MQCONN** o **MQCONNX** intentan entonces utilizar el canal que ha definido, a menos que se altere temporalmente el canal en una estructura MQCD referenciada desde la estructura MQCNO suministrada a **MQCONNX**.

Tareas relacionadas

“Utilización de las variables de entorno de IBM MQ” en la página 66
Puede utilizar mandatos para visualizar los valores actuales o restablecer los valores de las variables de entorno de IBM MQ.

“Creación de un canal de conexión de cliente en IBM MQ MQI client utilizando MQCNO” en la página 43
Puede definir un canal de conexión de cliente en la estación de trabajo cliente utilizando la estructura MQCNO en una llamada MQCONNX.

Cambio del puerto predeterminado TCP/IP

De forma predeterminada, para TCP/IP, IBM MQ presupone que el canal se conectará al puerto 1414. Si es necesario, puede cambiar el valor predeterminado.

Acerca de esta tarea

Puede cambiar el número de puerto utilizando una de las tres opciones siguientes:

- Utilizando la variable de entorno **MQSERVER** .
- Cambiando el archivo `mqclient.ini` .
- Añadiendo IBM MQ al archivo de servicios.

Procedimiento

- Para cambiar el número de puerto utilizando la variable de entorno **MQSERVER** , añada el número de puerto entre corchetes como la última parte de *ConnectionName*, por ejemplo:
  - Linux AIX En AIX and Linux:

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

  - Windows En Windows:

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

- Para cambiar el número de puerto utilizando el archivo `mq.ini` , añada el número de puerto al nombre de protocolo, por ejemplo:

  ```
  TCP:
  port=2001
  ```

- Para cambiar el número de puerto añadiendo IBM MQ al archivo de servicios, realice los pasos descritos en "Utilización del escucha TCP/IP en AIX and Linux" en la página 283.

Windows *Cambio del socket predeterminado SPX*

De forma predeterminada, para SPX, IBM MQ presupone que el canal se conectará al socket 5E86. Si es necesario, puede cambiar el valor predeterminado.

Acerca de esta tarea

Puede cambiar el número de puerto utilizando una de las opciones siguientes:

- Utilizando la variable de entorno **MQSERVER** .

  Para las conexiones SPX, especifique el ConnectionName y el socket con el formato `network.node(socket)`. Si el cliente y el servidor de IBM MQ están en la misma red, no es necesario especificar la red. Si está utilizando el socket predeterminado, no es necesario especificar el socket.
- Cambiando la stanza SPX del archivo `mqclient.ini` file.Changing el archivo `qm.ini` .

Procedimiento

- Para cambiar el número de puerto para una conexión SPX utilizando la variable de entorno **MQSERVER** , especifique *ConnectionName* y el socket con el formato `network.node(socket)` tal como se muestra en el ejemplo siguiente:

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

  Nota: Si el cliente y el servidor de IBM MQ están en la misma red, no es necesario que especifique la red. Si está utilizando el socket predeterminado, no es necesario que especifique el socket.
- Para cambiar el número de puerto utilizando el archivo `qm.ini` , añada el número de puerto al nombre de protocolo, por ejemplo:

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

Creación de un canal de conexión de cliente en IBM MQ MQI client utilizando MQCNO

Puede definir un canal de conexión de cliente en la estación de trabajo cliente utilizando la estructura MQCNO en una llamada MQCONNX.

Acerca de esta tarea

Una aplicación IBM MQ MQI client puede utilizar la estructura de opciones de conexión, MQCNO, en una llamada **MQCONNX** para hacer referencia a una estructura de definición de canal, MQCD, que contiene la definición de un canal de conexión con el cliente.

De este modo, la aplicación cliente puede especificar los atributos **ChannelName**, **TransportType** y **ConnectionName** de un canal en tiempo de ejecución, que permite a la aplicación cliente conectarse a varios gestores de colas de servidor al mismo tiempo.

Tenga en cuenta que si se define un canal utilizando la variable de entorno **MQSERVER**, no es posible especificar los atributos **ChannelName**, **TransportType** y **ConnectionName** en tiempo de ejecución.

Una aplicación cliente también puede especificar atributos de un canal, como por ejemplo **MaxMsgLength** y **SecurityExit**. Especificar dichos atributos permite a la aplicación cliente especificar valores para los atributos que no sean los valores predeterminados y permite que se llame a los programas de salida de canal del extremo del cliente de un canal MQI.

Si un canal utiliza o TLS (seguridad de la capa de transporte), una aplicación cliente también puede proporcionar información relacionada con TLS en la estructura MQCD. Se puede proporcionar información adicional relacionada con TLS en la estructura de opciones de configuración TLS, MQSCO, a la que también se hace referencia en la estructura MQCNO de una llamada **MQCONNX**.

Para obtener más información sobre las estructuras MQCNO, MQCD y MQSCO, consulte MQCNO, MQCD y MQSCO.

Nota: El programa de ejemplo para MQCONNX se denomina **amqscnxc**. Otro programa de ejemplo denominado **amqsssl c** muestra el uso de la estructura MQSCO.

Tareas relacionadas

“Creación de un canal de conexión de cliente en IBM MQ MQI client utilizando MQSERVER” en la página 38

Puede definir un canal de conexión de cliente en una estación de trabajo cliente utilizando la variable de entorno **MQSERVER** .

Creación de definiciones de conexión de servidor y de conexión de cliente en el servidor

Puede crear ambas definiciones en el servidor y, a continuación, poner la definición de conexión de cliente a disposición del cliente.

Acerca de esta tarea

En primer lugar, se define un canal de conexión de servidor y luego se define un canal de conexión de cliente:

- En todas las plataformas, puede utilizar los mandatos de IBM MQ Script (MQSC), mandatos de formato de mandato programable (PCF) para definir un canal de conexión de servidor en la máquina de servidor.
- Windows Linux En Linux y Windows, también puede utilizar IBM MQ Explorer.
- z/OS En z/OS, también puede utilizar los paneles de Operación y Control.
- IBM i En IBM i también puede utilizar la interfaz de panel.

Las definiciones de canal de conexión de cliente creadas en el servidor se ponen a disposición de los clientes utilizando una tabla de definiciones (CCDT).

Procedimiento

1. Para definir un canal de conexión de servidor, consulte “Definición del canal de conexión del servidor en el servidor” en la página 57.
2. Para definir un canal de conexión de cliente, consulte “Definición del canal de conexión de cliente en el servidor” en la página 58.

Tareas relacionadas

“Configuración de una tabla de definición de canal de cliente en formato binario” en la página 45
La tabla de definición de canal de cliente (CCDT) determina las definiciones de canal y la información de autenticación que utilizan las aplicaciones cliente para poder conectarse al gestor de colas. En Multiplataforms, se crea automáticamente una tabla de definición de canal de cliente binaria que contiene valores predeterminados cuando se crea el gestor de colas. Puede utilizar el mandato **runmqsc** para actualizar una tabla de definición de canal de cliente binaria.

“Definición del canal de conexión del servidor en el servidor” en la página 57
Cree un canal de conexión de servidor para el gestor de colas.

“Definición del canal de conexión de cliente en el servidor” en la página 58
Después de haber definido el canal de conexión con el servidor, puede definir el canal de conexión con el cliente correspondiente.

“Acceso a las definiciones de canal de conexión de cliente” en la página 59
Puede dejar la tabla de definición de canal de cliente (CCDT) disponible para las aplicaciones cliente copiándola o compartiéndola y, a continuación, especifique la ubicación y el nombre del sistema cliente. También puede localizar una tabla de definición de canal de cliente (CCDT) a través de un URL.

Configuración de tablas de definición de canal de cliente

Una tabla de definición de canal de cliente (CCDT) define los canales de conexión de cliente y sus atributos. Los clientes leen este archivo para determinar a qué gestores de colas se van a conectar. El archivo CCDT puede tener formato JSON o binario.

Acerca de esta tarea

El gestor de colas no lee el archivo CCDT. Solo se utiliza para proporcionar las definiciones de canal y la información de autenticación a los clientes.

Cuando se crea un gestor de colas, se crea automáticamente una tabla de definición de canal de cliente en formato binario. Puede actualizar las definiciones de canal de cliente almacenadas en esta tabla utilizando solo el mandato **runmqsc**.

Una tabla de definición de canal de cliente en formato JSON es un archivo de texto plano con un formato .json. Puede crear y actualizar esta tabla manualmente, lo que es menos restringido que usar el mandato **runmqsc**.

z/OS Los clientes de z/OS JMS que se ejecutan dentro de un servidor de aplicaciones utilizan una CCDT para hacer referencia a detalles de conexión de gestor de colas remoto. A partir de IBM MQ for z/OS 9.1, IBM MQ Advanced for z/OS Value Unit Edition permite que los clientes de JMS se conecten de forma remota a los gestores de colas en otras particiones lógicas de z/OS. Por lo tanto, estos clientes también pueden utilizar tablas de definición de canal de cliente.

Para ayudarle a configurar las tablas de definición de canal de cliente para que funcionen con sus clientes, elija entre las tareas siguientes:

Procedimiento

- “Configuración de una tabla de definición de canal de cliente en formato binario” en la página 45
- “Configuración de una tabla de definición de canal de cliente en formato JSON” en la página 47
- “Ubicaciones para la tabla de definición de canal de cliente” en la página 54
- “Acceso de URL a la tabla de definición de canal de cliente” en la página 55

Conceptos relacionados

Cliente MQI: Tabla de definición de canal de cliente (CCDT)

Tareas relacionadas

“Configuración de un clúster uniforme” en la página 431

Los clústeres uniformes permiten que se diseñen las aplicaciones para la escala y la disponibilidad, y se puedan conectar a cualquiera de los gestores de colas dentro de ese clúster uniforme.

Configuración de una tabla de definición de canal de cliente en formato binario

La tabla de definición de canal de cliente (CCDT) determina las definiciones de canal y la información de autenticación que utilizan las aplicaciones cliente para poder conectarse al gestor de colas. En Multiplataforms, se crea automáticamente una tabla de definición de canal de cliente binaria que contiene valores predeterminados cuando se crea el gestor de colas. Puede utilizar el mandato **runmqsc** para actualizar una tabla de definición de canal de cliente binaria.

Antes de empezar

También puede crear una CCDT en formato JavaScript Object Notation (JSON), y el uso de este formato alternativo tiene algunas ventajas sobre el uso de una CCDT binaria. Consulte “Configuración de una tabla de definición de canal de cliente en formato JSON” en la página 47.

Los clientes en todas las plataformas pueden ver y utilizar las tablas de definición de canal de cliente. Sin embargo, la tabla de definición de canal de cliente binaria solo se puede crear y modificar bajo IBM MQ for Multiplatforms.

Acerca de esta tarea

Multi En Multiplatforms:

- Una CCDT binaria se crea automáticamente en el directorio `@ipcc` bajo el directorio de datos para el gestor de colas.
- Además de crearse automáticamente, la tabla de definición de canal de cliente binaria asociada a un gestor de colas se mantiene en sincronización con las definiciones de objeto. Cuando define, modifica o suprime un objeto de canal cliente, tanto la definición de objeto de gestor de colas como la entrada en la tabla de definición de canal de cliente se actualizan como parte de la misma operación.

  Notas:

  – El diseño del archivo CCDT de IBM MQ es que el archivo CCDT se ha reducido, solo después de que todos los canales de conexión de cliente definidos por el usuario se hayan definido realmente. Cuando se suprime un canal de conexión cliente, solo se marca como suprimido en el archivo CCDT, pero no se elimina físicamente.

  – Para obligar al archivo CCDT a reducirse, después de suprimir uno o más canales de conexión de cliente, emita el mandato siguiente:

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

- Puede utilizar el mandato **runmqsc** para cambiar la ubicación y el contenido de la tabla de definición de canal de cliente binaria.

Los clientes en todas las plataformas pueden ver y utilizar las tablas de definición de canal de cliente binarias.

Procedimiento

- Multi

  Cree una tabla de definición de canal de cliente binaria predeterminada.

  En Multiplatforms, se crea una CCDT binaria predeterminada denominada `AMQCLCHL.TAB` cuando se crea un gestor de colas.

De forma predeterminada, el archivo AMQCLCHL.TAB está ubicado en el directorio siguiente de un servidor:

– IBM i En IBM i, en el sistema de archivos integrado:

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

– Linux AIX En sistemas AIX and Linux:

/prefix/qmgrs/*QUEUEMANAGERNAME*/@ipcc

El nombre del directorio al que hace referencia *QUEUEMANAGERNAME* distingue entre mayúsculas y minúsculas en los sistemas AIX and Linux . Puede que el nombre del directorio no sea el mismo que el nombre del gestor de colas, si el nombre del gestor de colas tiene caracteres especiales.

– Windows En Windows:

MQ_INSTALLATION_PATH\data\qmgrs*QUEUEMANAGERNAME*\@ipcc

donde *MQ_INSTALLATION_PATH* representa el directorio de alto nivel en el que está instalado IBM MQ.

No obstante, puede que haya elegido un directorio diferente para los datos del gestor de colas. Puede especificar el parámetro **-md** *DataPath* cuando utilice el mandato **crtmqm** . Si lo hace, AMQCLCHL.TAB se encuentra en el directorio @ipcc de *DataPath* que ha especificado.

- Localice la tabla de definición de canal de cliente.
  – En el sistema del cliente
  – En una ubicación compartida por más de un cliente
  – En el servidor como un archivo compartido

  Consulte “Ubicaciones para la tabla de definición de canal de cliente” en la página 54.

  a) Cree una tabla de definición de canal de cliente binaria predeterminada en una máquina cliente.
     – Utilice el mandato runmqsc con el parámetro **-n**.
     – La tabla de definición de canal de cliente se crea en la ubicación indicada por **MQCHLLIB** y con el nombre de archivo indicado por **MQCHLTAB**, que es AMQCLCHL.TAB de forma predeterminada.
     – **Importante:** Si especifica el parámetro **-n**, no debe especificar ningún otro parámetro.

  b) Cambie la ubicación.

     Puede cambiar la vía de acceso a la tabla de definición de canal de cliente estableciendo **MQCHLLIB**. Se debe tener en cuenta que, si tiene varios gestores de colas en el mismo servidor, compartirán la misma ubicación de tabla de definición de canal de cliente.

- Acceda a la tabla de definición de canal de cliente

  Puede acceder a la tabla de definición de canal de cliente:
  – De forma remota desde un archivo, ftp o URL http, definiendo la variable de entorno **MQCCDTURL**.
  – Localmente estableciendo las variables de entorno **MQCHLLIB** y **MQCHLTAB**.
  – Localmente definiendo los atributos **ChannelDefinitionDirectory** y **ChannelDefinitionFile** de la stanza CHANNELS en el archivo de configuración del cliente.

  Consulte “Ubicaciones para la tabla de definición de canal de cliente” en la página 54 para ver varios ejemplos.

- Consulte o edite el contenido de la tabla de definición de canal de cliente.

  Puede ver el contenido de la tabla de definición de canal de cliente con el mandato **runmqsc** :

  1. Establezca las variables de entorno en Acceder a la tabla de definición de canal de cliente
  2. Ejecute el mandato `runmqsc -n`

3. Ejecute el mandato DISPLAY CHANNEL(*), por ejemplo

Multi En Multiplatforms, puede editar también el contenido de la tabla de definición de canal de cliente binaria mediante el mandato **runmqsc**. Cada entrada de una CCDT representa una conexión de cliente a un gestor de colas específico. Se añade una nueva entrada cuando se define un canal de conexión de cliente mediante el mandato **DEFINE CHANNEL**, y la entrada se actualiza cuando se modifican los canales de conexión de cliente mediante el mandato **ALTER CHANNEL**. Consulte **runmqsc** para obtener más ejemplos de cómo utilizar el mandato.

- Proporcione a los clientes la información de autenticación para comprobar la revocación de certificados TLS.
  a) Defina una lista de nombres que contenga objetos de información de autenticación.
  b) Establezca el atributo de gestor de colas **SSLCRLNL** en el nombre de la lista de nombres.

Conceptos relacionados

Trabajar con certificados revocados

Tareas relacionadas

"Configuración de una tabla de definición de canal de cliente en formato JSON" en la página 47

La tabla de definición de canal de cliente (CCDT) determina las definiciones de canal y la información de autenticación que utilizan las aplicaciones cliente para poder conectarse al gestor de colas. Puede utilizar un editor de texto para crear y actualizar una tabla de definición de canal de cliente JSON (JavaScript Object Notation).

Configuración de una tabla de definición de canal de cliente en formato JSON

La tabla de definición de canal de cliente (CCDT) determina las definiciones de canal y la información de autenticación que utilizan las aplicaciones cliente para poder conectarse al gestor de colas. Puede utilizar un editor de texto para crear y actualizar una tabla de definición de canal de cliente JSON (JavaScript Object Notation).

Antes de empezar

Multi Si utiliza IBM MQ for Multiplatforms, en su lugar puede utilizar la tabla de definición de canal de cliente binaria que se crea automáticamente al crear un gestor de colas. Consulte "Configuración de una tabla de definición de canal de cliente en formato binario" en la página 45.

Acerca de esta tarea

El nombre de archivo del esquema CCDT para el formato JSON es:

Linux

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

Windows

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

No hay ninguna tabla de definición de canal de cliente JSON predeterminada y IBM MQ no proporciona ninguna herramienta para crear o editar las tablas de definición de canal de cliente en formato JSON. Sin embargo, tiene más opciones de configuración al desarrollar manualmente una tabla de definición de canal de cliente JSON que cuando utiliza el mandato **runmqsc** para trabajar con una tabla de definición de canal de cliente binaria:

- No es necesario que esté utilizando IBM MQ for Multiplatforms para crear y editar un archivo de tabla de definición de canal de cliente JSON.
- Al utilizar el formato JSON, puede definir definiciones de canal duplicadas con el mismo nombre. Al desplegar IBM MQ en la nube, puede utilizar esto para hacer que su despliegue sea escalable y esté altamente disponible.
- El archivo JSON es legible por el usuario, lo que puede simplificar la configuración del gestor de colas.
- Un formato de archivo plano se puede integrar con:

  - Un conjunto de herramientas de control de versiones para realizar el seguimiento del historial de tablas de definición de canal de cliente
  - Un conjunto de herramientas de automatización de entrega continua
- No es necesario que el conjunto de herramientas especializadas mantengan el archivo de tabla de definición de canal de cliente.
- El archivo es más pequeño.
- Este formato proporciona compatibilidad con versiones anteriores y posteriores.

Notas:

1. El estándar JSON considera las claves duplicadas como válidas, sin embargo, el analizador JSON solo toma el último valor leído de claves duplicadas al asignar los atributos. Por lo tanto, al definir canales duplicados, cada canal debe ser un elemento de un valor de matriz asignado a la clave '`channel`'.
2. Las tablas de definición de canal de cliente JSON no admiten el almacenamiento de ubicaciones de servidor LDAP (Lightweight Directory Access Protocol) para la información de ubicación del programa de respuesta de las listas de revocación de certificados (CRL) y el protocolo de estado de certificados en línea (OCSP).

Tabla 7. Requisitos de codificación por plataforma

Plataforma	**Codificación de cliente JMS**	**Codificación de cliente C**
IBM i IBM i	ASCII	EBCDIC
ALW AIX, Linux, and Windows	ASCII	ASCII
z/OS z/OS	ASCII o EBCDIC	No aplicable

Atención: Cuando proporciona cualquier definición para un canal a través de una definición de canal de cliente JSON (incluyendo una *dispersa* que no incluye todos los atributos), se construye una definición de canal completa con todos los atributos definidos, utilizando los valores predeterminados para cualquier cosa no especificada en el JSON.

Por lo tanto, debe proporcionar valores específicos para cada atributo para el que no desee el valor predeterminado.

Procedimiento

- Cree una tabla de definición de canal de cliente JSON
  a) Cree un archivo sin formato con una extensión `.json` con un editor de texto genérico.
  b) Defina una tabla de definición de canal de cliente.

  Consulte “Ejemplos de tabla de definición de canal de cliente de JSON” en la página 51 y “Atributos de canal que admite la tabla de definición de canal de cliente JSON” en la página 49.
- Localice la tabla de definición de canal de cliente.
  - En el sistema del cliente
  - En una ubicación compartida por más de un cliente
  - En el servidor como un archivo compartido

  Consulte “Ubicaciones para la tabla de definición de canal de cliente” en la página 54.
- Valide una tabla de definición de canal de cliente JSON

  Valide la tabla de definición de canal de cliente con el esquema con un enlazador JSON.

  Consulte Cómo validar un archivo JSON de CCDT de IBM MQ con el esquema para obtener información sobre cómo crear un archivo CCDT con dos canales y validar que funciona.

El esquema de tabla de definición de canal de cliente se incluye con los paquetes de producto y cliente:

  - Linux AIX En sistemas AIX and Linux:

    `$MQ_INSTALLATION_PATH/lib` y `/lib` en los paquetes de producto y cliente, respectivamente.

  - Windows En Windows:

    `%MQ_INSTALLATION_PATH%\bin` y `\bin` en los paquetes de producto y cliente, respectivamente.

  Notas:

  - Los enlazadores JSON están disponibles en línea.
  - El esquema define atributos obligatorios con la clave '`required`'.
  - El esquema define tipos de datos de atributo con la clave '`type`'.

- Acceda a la tabla de definición de canal de cliente

  Puede acceder a la tabla de definición de canal de cliente:

  - De forma remota desde un archivo, ftp o URL http, definiendo la variable de entorno **MQCCDTURL**.
  - Localmente estableciendo las variables de entorno **MQCHLLIB** y **MQCHLTAB**.
  - Localmente definiendo los atributos **ChannelDefinitionDirectory** y **ChannelDefinitionFile** de la stanza CHANNELS en el archivo de configuración del cliente.

  Consulte “Ubicaciones para la tabla de definición de canal de cliente” en la página 54 para ver varios ejemplos.

- Consulte o edite el contenido de la tabla de definición de canal de cliente

  Cada entrada de una CCDT representa una conexión de cliente a un gestor de colas específico. Puede ver o editar el contenido de la tabla de definición de canal de cliente con un editor de texto.

  Si solo desea ver la tabla de definición de canal de cliente, también puede hacerlo utilizando el mandato **runmqsc** como se indica a continuación:

  1. Establezca las variables de entorno para proporcionarle acceso a la tabla de definición de canal de cliente, tal como se describe en el paso anterior.
  2. Ejecute el mandato `runmqsc -n` . Para obtener más información, consulte runmqsc.
  3. Ejecute el mandato **DISPLAY CHANNEL**. Por ejemplo, ejecute `DISPLAY CHANNEL(*)`.

Conceptos relacionados

Trabajar con certificados revocados

Tareas relacionadas

“Configuración de una tabla de definición de canal de cliente en formato binario” en la página 45
La tabla de definición de canal de cliente (CCDT) determina las definiciones de canal y la información de autenticación que utilizan las aplicaciones cliente para poder conectarse al gestor de colas. En Multiplataforms, se crea automáticamente una tabla de definición de canal de cliente binaria que contiene valores predeterminados cuando se crea el gestor de colas. Puede utilizar el mandato **runmqsc** para actualizar una tabla de definición de canal de cliente binaria.

“Configuración de un clúster uniforme” en la página 431
Los clústeres uniformes permiten que se diseñen las aplicaciones para la escala y la disponibilidad, y se puedan conectar a cualquiera de los gestores de colas dentro de ese clúster uniforme.

Atributos de canal que admite la tabla de definición de canal de cliente JSON
Una lista de los atributos de canal de conexión cliente que admite la tabla de definición de canal de cliente JSON. Esta lista es un subconjunto de los atributos que admite la tabla de definición de canal de cliente binaria.

Correlación de atributos

Estos atributos se insertan en el objeto de canal siguiente:

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

donde `$CHANNEL_X_KEY_VALUE_LIST` es una lista separada por comas de los atributos enumerados en la tabla siguiente.

Consulte "Ejemplos de tabla de definición de canal de cliente de JSON" en la página 51 para ver los casos de uso básico.

El esquema JSON se suministra en `/opt/mqm/lib/ccdt_schema.json`. Para descubrir qué valores son válidos para cada uno de los atributos, consulte el esquema JSON.

En la tabla siguiente se enumera el objeto JSON, la clave y el tipo de datos, junto con la definición de atributo de canal binario correspondiente.

Atención: los atributos necesarios son **name** y **type** del canal. Si define también **portRange**, son necesarios también los atributos *low* y *high*.

Objeto JSON	Clave JSON	Tipo de datos JSON	Definición de atributo binario
channel (matriz)	nombre	STRING	CHANNEL
channel (matriz)	Tipo	STRING	CHLTYPE
channel.clientConnection	queueManager	STRING	QMNAME
channel.clientConnection.connection (matriz)	host	STRING	CONNAME
channel.clientConnection.connection	port	INT	CONNAME
channel.compression.header (matriz)	cabecera	STRING	COMPHDR
channel.compression.message (matriz)	mensaje	STRING	COMPMSG
channel.connectionManagement	affinity	STRING	AFINIDAD
channel.connectionManagement	clientWeight	INT	CLNTWGHT
channel.connectionManagement	defaultReconnect	STRING	DEFRECON
channel.connectionManagement	disconnectInterval	INT	DISCINT
channel.connectionManagement	heartInterval	INT	HBINT
channel.connectionManagement	keepAliveInterval	INT	KAINT
channel.connectionManagement	sharingConversations	INT	SHARECNV
channel.connectionManagement.localAddress (matriz)	host	STRING	LOCLADDR
channel.connectionManagement.localAddress (matriz)	port	INT	LOCLADDR
channel.connectionManagement.localAddress.portRange	alto	INT	LOCLADDR
channel.connectionManagement.localAddress.portRange	low	INT	LOCLADDR
channel.exits.receive (matriz)	nombre	STRING	RCVEXIT
channel.exits.receive (matriz)	userData	STRING	RCVDATA

Objeto JSON	Clave JSON	Tipo de datos JSON	Definición de atributo binario
channel.exits.security	nombre	STRING	SCYEXIT
channel.exits.security	userData	STRING	SCYDATA
channel.exits.send (matriz)	nombre	STRING	SENDEXIT
channel.exits.send (matriz)	userData	STRING	SENDDATA
channel.general	description	STRING	DESCR
channel.general	maximumMessageLength	INT	MAXMSGL
channel.timestamps	altered	STRING	ALTDATE y ALTTIME
channel.transmissionSecurity	certificateLabel	STRING	CERTLABL
channel.transmissionSecurity	certificatePeerName	STRING	SSLPEER
channel.transmissionSecurity	cipherSpecification	STRING	SSLCIPH

Notas:

- `channel.connectionManagement.localAddress` se puede definir como una de las combinaciones siguientes de claves:
  - host y port
  - host y portRange
  - port
  - portRange
- La clave JSON `channel.timestamps altered` es opcional y, si no se ha definido, el valor toma de forma predeterminada la última hora modificada del archivo CCDT de JSON. Sin embargo, si el entorno se ha configurado para captar la CCDT de un URL, el valor predeterminado es la hora cuando se descargó por última vez el archivo.
- `channel.clientConnection.connection` debe incluir las claves host y port.
- La clave altered es una serie única que encapsula los atributos ALTDATE y ALTTIME.
- El tipo de transporte solo puede ser TCP, por lo tanto, los atributos siguientes no están definidos en el esquema:
  - **TRPTYPE**
  - **USERID**
  - **PASSWORD**
  - **MODENAME**
  - **TPNAME**

Referencia relacionada

Atributos de canal para tipos de canal

Ejemplos de tabla de definición de canal de cliente de JSON

Utilice los ejemplos que se listan en este tema como base para los requisitos.

Abra un editor de texto genérico y copie uno de los ejemplos siguientes:

- "Definir una conexión de cliente simple" en la página 52
- "Definir un canal y un gestor de colas utilizando TLS" en la página 52

- “Definir un canal y un gestor de colas que no utilizan TLS” en la página 52
- “Definir dos canales con el mismo nombre” en la página 53
- “Lista completa de definiciones de atributos de canal CCDT para un canal de conexión de cliente” en la página 53

Definir una conexión de cliente simple

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

Definir un canal y un gestor de colas utilizando TLS

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

Definir un canal y un gestor de colas que no utilizan TLS

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

```
      },
      "type": "clientConnection"
    }
  ]
}
```

Definir dos canales con el mismo nombre

Cada canal se conecta a dos gestores de colas distintos:

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

Lista completa de definiciones de atributos de canal CCDT para un canal de conexión de cliente

```
{
  "channel":
  [
    {
      "compression":
      {
        "header": [ "system" ],
        "message": [ "zlibfast" ]
      },
      "connectionManagement":
      {
        "sharingConversations": 10,
        "clientWeight": 1,
        "affinity": "none",
        "defaultReconnect": "yes",
        "heartbeatInterval": 600,
        "keepAliveInterval": -1,
        "localAddress":
```

```
        [
          {
            "portRange":
            {
              "low": 2020,
              "high": 3030
            }
          }
        ]
      },
      "exits":
      {
        "receive":
        [
          {
          "name": "",
          "userData": ""
          }
        ],
        "security":
        {
          "name": "",
          "userData": ""
        },
        "send":
        [
           {
             "name": "",
             "userData": ""
           }
        ]
      },
      "general":
      {
        "description": "First channel",
        "maximumMessageLength": 4194304
      },
      "name": "the_channel",
      "clientConnection":
      {
        "connection":
        [
          {
            "host": "localhost",
            "port": 1414
          }
        ],
        "queueManager": "QM1"
      },
      "timestamps":
      {
        "altered": "2018-12-04T15:37:22.000Z"
      },
      "transmissionSecurity":
      {
        "cipherSpecification": "",
        "certificateLabel": "",
        "certificatePeerName": ""
      },
      "type": "clientConnection"
    }
  ]
}
```

Referencia relacionada

Atributos de canal para tipos de canal

Atributos de canal en orden alfabético

Ubicaciones para la tabla de definición de canal de cliente

IBM MQ permite recuperar una tabla de definición de canal de cliente a partir de un archivo, FTP o un URL HTTP. Puede hacer que la CCDT sea accesible para el cliente como un archivo compartido, mientras sigue ubicada en el servidor. O bien, puede distribuir la tabla de definición de canal de cliente, copiando dicha tabla en los sistemas cliente individuales o copiándola en una ubicación compartida entre más de un cliente.

Si utiliza FTP para copiar el archivo, recuerde utilizar la opción `bin` para establecer la modalidad binaria; no utilice la modalidad `ASCII` predeterminada. Independientemente del método que elija para hacer que la CCDT esté disponible, la ubicación debe ser segura, para impedir que se efectúen cambios no autorizados en los canales.

Cómo alojar el archivo CCDT en un servidor

La CCDT se puede alojar en una ubicación central a la que se puede acceder a través de un URL, eliminando la necesidad de actualizar individualmente la CCDT para cada cliente desplegado. Las aplicaciones nativas (C/C++, COBOL y RPG) y .NET no gestionadas pueden extraer la CCDT de un URL, ya sea un archivo local, FTP o recurso HTTP.

El comportamiento de almacenamiento en memoria caché predeterminado de los clientes de IBM MQ es que un archivo CCDT sólo se extraiga si la hora de modificación de archivo es diferente de la última hora a la que se ha recuperado. Al igual que con la mayoría de las opciones de configuración de cliente, existen diversas maneras en que se puede proporcionar la ubicación de URL:

- **CCDTUrlPtr** y **CCDTUrlOffset** a través de la estructura MQCNO que se pasa a la llamada de MQI MQCONNX
- Variable de entorno **MQCCDTURL**
- Atributo **ChannelDefinitionDirectory** de la stanza Channels de `mqclient.ini`

Se soportan los URL autenticados y no autenticados. A continuación, se detallan algunos ejemplos:

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

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

Si desea utilizar este soporte con FTP o HTTP, todavía tiene que alojar el archivo CCDT en un servidor, pero todas las aplicaciones cliente pueden recoger automáticamente los cambios en las definiciones de canal sin enviar manualmente las actualizaciones o tener que montar un sistema de archivos en red en cada cliente. Para obtener más información, consulte “Acceso de URL a la tabla de definición de canal de cliente” en la página 55.

Cómo especificar la ubicación de la CCDT en el cliente

En un sistema cliente, puede especificar la ubicación de la CCDT de las siguientes maneras:

- Utilizando las variables de entorno **MQCHLLIB** para especificar el directorio donde se encuentra la tabla, y **MQCHLTAB** para especificar el nombre de archivo de la tabla.
- Mediante el archivo de configuración de cliente. En la stanza CHANNELS, utilice el atributo **ChannelDefinitionDirectory** para especificar el directorio donde se encuentra la tabla y el atributo **ChannelDefinitionFile** para especificar el nombre de archivo.
- Al proporcionar un URL (archivo, FTP o HTTP) para una tabla de definición de canal de cliente que se aloja en una ubicación central como se describe anteriormente.

Si se especifica la ubicación en el archivo de configuración de cliente y también mediante las variables de entorno, las variables de entorno tienen prioridad. Puede utilizar esta característica para especificar una ubicación estándar en el archivo de configuración del cliente, y alterarla temporalmente mediante las variable de entorno, cuando sea necesario.

Si utiliza un URL para proporcionar la ubicación de la CCDT, el orden de prioridad para una aplicación cliente nativa para encontrar la definición de canal de cliente es tal como se describe en “Acceso de URL a la tabla de definición de canal de cliente” en la página 55.

Acceso de URL a la tabla de definición de canal de cliente

Puede alojar una tabla de definición de canal de cliente (CCDT) en una ubicación central a la que se puede acceder a través de un URL, lo que elimina la necesidad de actualizar individualmente la tabla de definición de canal de cliente para cada cliente desplegado.

Un URL de definición de canal de cliente se puede localizar mediante un URL de los modos siguientes:

- Mediante la programación utilizando MQCNO
- Mediante el uso de variables de entorno

Atención: Puede utilizar la opción de variable de entorno para proporcionar el URL sólo para programas nativos que se conectan como clientes, es decir, aplicaciones C, COBOL o C++. Las variables de entorno tienen efecto en aplicaciones Java, JMS o .NET gestionadas.

IBM MQ permite recuperar una tabla de definición de canal de cliente a partir de un archivo, FTP o un URL HTTP.

- Utilizando la stanza CHANNELS de archivo `mqclient.ini`.

La variable de entorno **MQCCDTURL** le permite proporcionar un URL de archivo, ftp o http como un único valor del que se puede obtener una tabla de definición de canal de cliente.

También puede utilizar la vía de acceso del directorio especificada por la variable de entorno **MQCHLLIB** (o la vía de acceso especificada por el atributo **ChannelDefinitionDirectory** en "Stanza CHANNELS del archivo de configuración de cliente" en la página 191) para localizar un archivo CCDT, ya sea a través del archivo, ftp o URL http, además del directorio del sistema de archivos local existente, es decir, `/var/mqm`). Tenga en cuenta que un valor **MQCHLLIB** es una raíz de directorio y funciona en combinación con **MQCHLTAB** para derivar el URL completo.

La autenticación básica en conexiones se admite a través de las credenciales que se cifran en el URL:

Conexiones autenticadas

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

Conexiones sin autenticar

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

Nota: Si desea utilizar conexiones autenticadas, deberá, al igual que con JMS, proporcionar el nombre de usuario y la contraseña cifrados en el URL.

El orden de prioridad para que una aplicación cliente nativa encuentre una definición de canal de cliente es ahora:

1. MQCD que proporciona **ClientConnOffset** y **ClientConnPtr** en MQCNO.
2. URL proporcionado por **CCDTUrlOffset** y **CCDTUrlPtr** en MQCNO.
3. Variable de entorno *MQSERVER* .
4. Si se define un archivo `mqclient.ini` y la stanza Channels contiene un atributo **ServerConnectionParms** , se utiliza el canal que define. Para obtener más información, consulte "Archivo de configuración de IBM MQ MQI client , mqclient.ini" en la página 172 y "Stanza CHANNELS del archivo de configuración de cliente" en la página 191.
5. Variable de entorno **MQCCDTURL** .
6. Variable de entorno **MQCHLLIB** y **MQCHLTAB** .
7. **ChannelDefinitionDirectory** y **ChannelDefinitionFile** en la "Stanza CHANNELS del archivo de configuración de cliente" en la página 191.

Importante: El acceso a un archivo CCDT utilizando un URL siempre abre una copia de sólo lectura del archivo, incluso cuando se utiliza el protocolo `file://`.

Al intentar abrir un archivo CCDT para acceso de grabación, por ejemplo cuando se utiliza el mandato MQSC de **DEFINE CHANNEL** desde un cliente, devuelve un mensaje de error que indica que el archivo no se ha podido abrir para acceso de grabación.

No obstante, es posible leer definiciones de información de autenticación y canal utilizando **runmqsc**.

Tareas relacionadas

"Acceso a las definiciones de canal de conexión de cliente" en la página 59
Puede dejar la tabla de definición de canal de cliente (CCDT) disponible para las aplicaciones cliente copiándola o compartiéndola y, a continuación, especifique la ubicación y el nombre del sistema cliente. También puede localizar una tabla de definición de canal de cliente (CCDT) a través de un URL.

"Configuración de una tabla de definición de canal de cliente en formato binario" en la página 45
La tabla de definición de canal de cliente (CCDT) determina las definiciones de canal y la información de autenticación que utilizan las aplicaciones cliente para poder conectarse al gestor de colas. En Multiplataforms, se crea automáticamente una tabla de definición de canal de cliente binaria que contiene valores predeterminados cuando se crea el gestor de colas. Puede utilizar el mandato **runmqsc** para actualizar una tabla de definición de canal de cliente binaria.

Utilización de una CCDT con IBM MQ classes for JMS

Referencia relacionada

CCDTURL
MQCNO - Opciones de conexión
XMSC_WMQ_CCDTURL

Windows Canales de conexión de cliente en Active Directory

En sistemas Windows que dan soporte a Active Directory, IBM MQ publica los canales de conexión de cliente en Active Directory para proporcionar enlaces dinámicos de cliente-servidor.

Cuando se definen objetos de canal de conexión de cliente, se graban en un archivo de definición de canal de cliente, denominado AMQCLCHL.TAB de forma predeterminada. Si los canales de conexión de cliente utilizan el protocolo TCP/IP, el servidor de IBM MQ también los publica en Active Directory. Cuando el cliente de IBM MQ determina cómo conectarse al servidor, busca una definición de objeto de canal de conexión de cliente relacionada utilizando el orden de búsqueda siguiente:

1. MQCONNX Estructura de datos MQCD
2. Variable de entorno de **MQSERVER**
3. Archivo de definiciones de canal de cliente
4. Active Directory

Este orden significa que cualquier aplicación actual no se ve afectada por los cambios. Puede considerar estas entradas en Active Directory como registros en el archivo de definición de canal de cliente y el cliente de IBM MQ las procesas de la misma forma. Para configurar y administrar el soporte para publicar definiciones de canales de conexión de cliente en Active Directory, utilice el mandato `setmqscp` tal como se describe en setmqscp.

Definición del canal de conexión del servidor en el servidor

Cree un canal de conexión de servidor para el gestor de colas.

Procedimiento

1. En la máquina servidor, defina un canal con el nombre que prefiera y un tipo de canal de *conexión con el servidor*.
   Por ejemplo:

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

2. Utilice el mandato siguiente para otorgar a la entrada acceso de conexión al gestor de colas:

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

   - Donde **SET CHLAUTH** utiliza el nombre del canal definido en el paso anterior.

- Donde *'dirección IP'* es la dirección IP del cliente.
- Donde *'ID usuario'* es el ID que desea proporcionar al canal para el control de accesos a la colas de destino. Este campo es sensible a las mayúsculas y minúsculas.

Puede elegir identificar la conexión de entrada mediante varios atributos distintos. En el ejemplo se utiliza la dirección IP. Entre los atributos alternativos se incluyen el ID de usuario del cliente y el nombre distinguido de asunto TLS. Para obtener más información, consulte Registros de autenticación de canal

Esta definición de canal se asocia al gestor de colas que se ejecuta en el servidor.

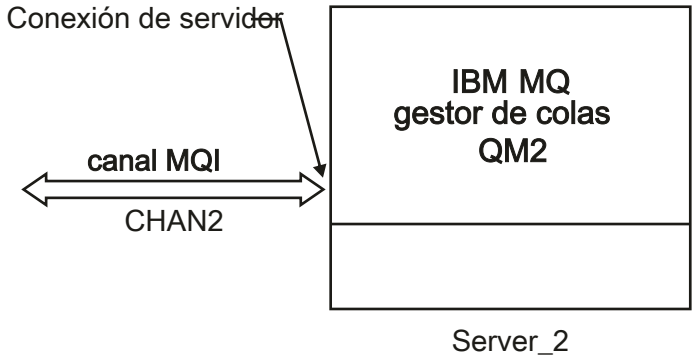


Figura 2. Definición del canal de conexión con el servidor

Tareas relacionadas

“Definición del canal de conexión de cliente en el servidor” en la página 58
Después de haber definido el canal de conexión con el servidor, puede definir el canal de conexión con el cliente correspondiente.

Definición del canal de conexión de cliente en el servidor

Después de haber definido el canal de conexión con el servidor, puede definir el canal de conexión con el cliente correspondiente.

Antes de empezar

Defina el canal de conexión del servidor. Para obtener más información, consulte “Definición del canal de conexión del servidor en el servidor” en la página 57.

Procedimiento

1. Defina un canal con el mismo nombre que el canal de conexión del servidor, pero un tipo de canal de *conexión de cliente*. Debe indicar el nombre de conexión (CONNAME). Para TCP/IP, el nombre de la conexión es la dirección de red o el nombre de host de la máquina servidor. También es conveniente especificar el nombre del gestor de colas (QMNAME) al que desea que se conecte la aplicación IBM MQ, que se ejecuta en el entorno de cliente. Al variar el nombre del gestor de colas, puede definir un conjunto de canales para conectarse a diferentes gestores de colas.

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

2. Utilice el mandato siguiente para otorgar a la entrada acceso de conexión al gestor de colas:

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

- Donde el mandato **SET CHLAUTH** utiliza el nombre del canal definido en el paso anterior.
- Donde *'dirección IP'* es la dirección IP del cliente.

- Donde *'ID usuario'* es el ID que desea proporcionar al canal para el control de accesos a la colas de destino. Este campo es sensible a las mayúsculas y minúsculas.

Puede elegir identificar la conexión de entrada mediante varios atributos distintos. En el ejemplo se utiliza la dirección IP. Entre los atributos alternativos se incluyen el ID de usuario del cliente y el nombre distinguido de asunto TLS. Para obtener más información, consulte Registros de autenticación de canal

Resultados

Multi En Multiplatforms, esta definición de canal se almacena en un archivo denominado tabla de definiciones de canal de cliente (CCDT), que se asocia al gestor de colas. La tabla de definiciones de canal de cliente puede contener más de una definición de canal de conexión de cliente.

Para obtener más información sobre la tabla de definiciones de canal de cliente, y la información correspondiente sobre cómo se almacenan las definiciones de canal de conexión de cliente en z/OS, consulte "Configuración de una tabla de definición de canal de cliente en formato binario" en la página 45.

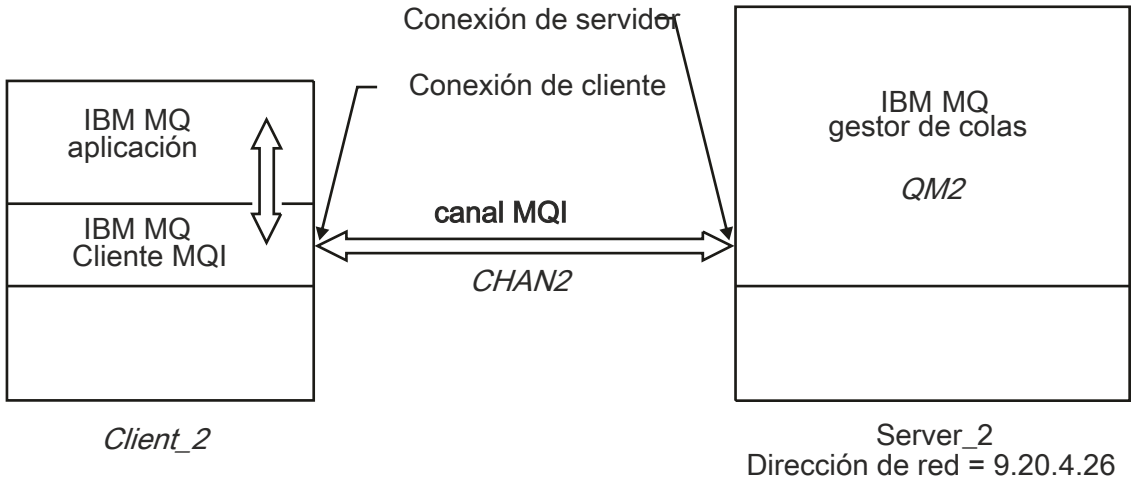


Figura 3. Definición de canal de conexión con el cliente

Referencia relacionada
DEFINE CHANNEL (definir un nuevo canal)
SET CHLAUTH (crear o modificar un registro de autenticación de canal)

Acceso a las definiciones de canal de conexión de cliente

Puede dejar la tabla de definición de canal de cliente (CCDT) disponible para las aplicaciones cliente copiándola o compartiéndola y, a continuación, especifique la ubicación y el nombre del sistema cliente. También puede localizar una tabla de definición de canal de cliente (CCDT) a través de un URL.

Antes de empezar

En esta tarea se supone que ha definido en una tabla de definición de canal de cliente los canales de conexión de cliente que necesita. Consulte "Configuración de tablas de definición de canal de cliente" en la página 44.

Acerca de esta tarea

Para que una aplicación cliente utilice la tabla de definición de canal de cliente (CCDT), la tabla CCDT debe estar disponible para la misma y especificar su ubicación y nombre. Existen varias formas de hacerlo:

- Puede copiar la CCDT en el sistema cliente.
- Puede copiar la CCDT en una ubicación compartida por más de un cliente.
- Puede hacer que la CCDT sea accesible para el cliente como un archivo compartido, mientras sigue ubicada en el servidor.

Las aplicaciones nativas de IBM MQ (C/C++, COBOL y RPG) y .NET no gestionadas pueden extraer la CCDT alojada en una ubicación central de un URL, ya sea un archivo local, ftp o recurso http.

Procedimiento

1. Deje la CCDT disponible paras las aplicaciones cliente de una de las siguientes formas:
   a) Opcional: Copie la CCDT en el sistema cliente.
   b) Opcional: Copie la CCDT en una ubicación compartida por más de un cliente.
   c) Opcional: Deje la CCDT en el servidor pero conviértala en compartible por el cliente.
   d) Opcional: Defina un archivo local, ftp o un URL http para una CCDT alojada en una ubicación central para que las aplicaciones nativas (C/C++, COBOL y RPG) y .NET no gestionadas puedan extraer la CCDT de este URL.

   Sea cual sea la ubicación que elija para la tabla CCDT, la ubicación debe ser segura para impedir que se realicen cambios no autorizados en los canales.
2. En el cliente, especifique la ubicación y el nombre del archivo que contiene la CCDT de una de las tres formas siguientes:
   a) Opcional: Utilice la stanza CHANNELS del archivo de configuración de cliente. Para obtener más información, consulte “Stanza CHANNELS del archivo de configuración de cliente” en la página 191.
   b) Opcional: Utilice las variables de entorno **MQCHLLIB** y **MQCHLTAB**.

      Por ejemplo, puede establecer las variables de entorno escribiendo:

      - Linux AIX En sistemas AIX and Linux:

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

      - IBM i En sistemas IBM i:

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

      donde *MQ_INSTALLATION_PATH* representa el directorio de alto nivel en el que está instalado IBM MQ.
   c) Opcional: Sólo en Windows, utilice el mandato de control **setmqscp** para publicar las definiciones de canal de conexión de cliente en Active Directory.
   d) Proporcione la ubicación de una CCDT alojada centralmente a través de un URL, ya sea mediante programación utilizando MQCNO, utilizando variables de entorno o utilizando las stanzas de archivo `mqclient.ini`. Para obtener más información, consulte “Ubicaciones para la tabla de definición de canal de cliente” en la página 54 y “Acceso de URL a la tabla de definición de canal de cliente” en la página 55.

   Si se establece la variable de entorno **MQSERVER** , un cliente IBM MQ utiliza la definición de canal de conexión de cliente especificada por **MQSERVER** en lugar de cualquier definición de la tabla de definiciones de canal de cliente.

Tareas relacionadas

“Configuración de una tabla de definición de canal de cliente en formato binario” en la página 45
La tabla de definición de canal de cliente (CCDT) determina las definiciones de canal y la información de autenticación que utilizan las aplicaciones cliente para poder conectarse al gestor de colas. En Multiplataforms, se crea automáticamente una tabla de definición de canal de cliente binaria que

contiene valores predeterminados cuando se crea el gestor de colas. Puede utilizar el mandato **runmqsc** para actualizar una tabla de definición de canal de cliente binaria.

Referencia relacionada

Cliente MQI: Tabla de definición de canal de cliente (CCDT)

ALW Programas de salida de canal para canales MQI

Hay tres tipos de salida de canal disponibles para el entorno de IBM MQ MQI client en AIX, Linux, and Windows.

Son las siguientes:

- Salida de emisión
- Salida de recepción
- Salida de seguridad

Estas salidas están disponibles tanto en el cliente como en el extremo del servidor del canal. Las salidas no están disponibles para la aplicación si utiliza la variable de entorno MQSERVER. Las salidas de canal se explican en Programas de salida de canal para canales de mensajería.

Las salidas de emisión y recepción funcionan conjuntamente. Hay varias maneras posibles en las que se pueden utilizar:

- Dividir y volver a montar un mensaje
- Comprimir y descomprimir datos de un mensaje (esta funcionalidad se proporciona como parte de IBM MQ, pero tal vez desee utilizar una técnica de compresión diferente)
- Cifrar y descifrar datos de usuario (esta funcionalidad se proporciona como parte de IBM MQ, pero tal vez desee utilizar una técnica de cifrado diferente)
- Registrar por diario cada mensaje enviado y recibido

Puede utilizar la salida de seguridad para asegurar que el cliente y el servidor de IBM MQ se identifiquen correctamente, y para controlar el acceso.

Si las salidas de emisión o recepción en el lado de conexión del servidor de la instancia de canal deben realizar llamadas MQI en la conexión con la que están asociadas, utilizan el manejador de conexión que se suministra en el campo MQCXP Hconn. Debe tener en cuenta que la conexión con el cliente y las salidas de envío y recepción no pueden efectuar llamadas MQI.

Conceptos relacionados

“Salidas de seguridad en una conexión con el cliente” en la página 62
Puede utilizar programas de salida de seguridad para verificar que la aplicación asociada en el otro extremo de un canal es genuina. Se aplican consideraciones especiales cuando se aplica una salida de seguridad a una conexión de cliente.

Salidas de usuario, salidas de API y servicios instalables de IBM MQ

Tareas relacionadas

Extensión de recursos del gestor de colas

Referencia relacionada

“Vía de acceso a las salidas” en la página 62
En el archivo de configuración del cliente se define una vía de acceso predeterminada para la ubicación de las salidas de canal. Las salidas de canal se cargan cuando se inicializa un canal.

“Identificación de la llamada API en un programa de salidas de envío o recepción” en la página 64
Cuando utilice canales MQI para cliente, el byte 10 del almacenamiento intermedio del agente identifica la llamada API en uso cuando se ha llamado a una salida de envío o recepción. Esto resulta útil para identificar qué flujos de canales incluyen datos de usuario y pueden necesitar que se procesen como, por ejemplo, la firma digital o el cifrado.

ALW Vía de acceso a las salidas

En el archivo de configuración del cliente se define una vía de acceso predeterminada para la ubicación de las salidas de canal. Las salidas de canal se cargan cuando se inicializa un canal.

En sistemas AIX, Linux, and Windows, se añade un archivo de configuración de cliente al sistema durante la instalación del IBM MQ MQI client. En ese archivo se define una vía de acceso predeterminada para la ubicación de las salidas de canal en el cliente, mediante la stanza:

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

donde *serie* es una ubicación de archivo en un formato adecuado para la plataforma

Cuando se inicializa un canal, después de una llamada MQCONN o MQCONNX, se busca el archivo de configuración de cliente. Se lee la stanza ClientExitPath y se cargan todas las salidas de canal especificadas en la definición de canal.

ALW Salidas de seguridad en una conexión con el cliente

Puede utilizar programas de salida de seguridad para verificar que la aplicación asociada en el otro extremo de un canal es genuina. Se aplican consideraciones especiales cuando se aplica una salida de seguridad a una conexión de cliente.

La Figura 4 en la página 63 ilustra el uso de las salidas de seguridad en una conexión de cliente, utilizando el gestor de autorizaciones sobre objetos de IBM MQ para autenticar un usuario.

El cliente establece el campo SecurityParmsPtr o SecurityParmsOffset en la estructura MQCNO y hay salidas de seguridad en ambos extremos del canal. Una vez que finaliza el intercambio de mensajes de seguridad normal y el canal está listo para ejecutarse, la estructura MQCSP se pasa a la salida de seguridad del cliente. La salida puede acceder a la estructura MQCSP utilizando el campo SecurityParms en la estructura MQCXP. El tipo de salida se establece en MQXR_SEC_PARMS. La salida de seguridad puede modificar las credenciales en la estructura MQCSP o dejarlas sin modificar.

A continuación, los datos que se devuelven de la salida se envían al extremo de conexión con el servidor del canal. La estructura MQCSP se reconstruye en el extremo de conexión de servidor del canal y se pasa a la salida de seguridad de conexión de servidor. La salida puede acceder a la estructura MQCSP utilizando el campo SecurityParms en la estructura MQCXP. La salida de seguridad recibe y procesa estos datos. Este proceso suele ser para invertir cualquier cambio realizado en las credenciales de la estructura MQCSP por la salida de cliente, que luego se utiliza para autorizar la conexión del gestor de colas. Se hace referencia a la estructura MQCSP resultante utilizando SecurityParmsPtr en la estructura MQCNO del sistema del gestor de colas.

La dirección de memoria que se devuelve con el campo SecurityParms de la estructura MQCXP debe permanecer direccionable y sin cambios hasta MQXR_TERM. Una salida no debe invalidar o liberar memoria en el sistema antes de que se invoque la salida para MQXR_TERM.

Si se establece el campo SecurityParmsPtr o SecurityParmsOffset en la estructura MQCNO y sólo hay una salida de seguridad en un extremo del canal, la salida de seguridad recibe y procesa la estructura MQCSP. Acciones como el cifrado son inadecuadas para una sola salida de usuario, ya que no hay salida para realizar la acción complementaria.

Si los campos SecurityParmsPtr y SecurityParmsOffset de la estructura MQCNO no están establecidos y hay una salida de seguridad en uno o ambos extremos del canal, se llama a la salida o salidas de seguridad. Cualquiera de las salidas de seguridad puede devolver su propia estructura MQCSP que se direcciona mediante el campo SecurityParmsPtr . La salida de seguridad no se vuelve a llamar hasta que termina (ExitReason de MQXR_TERM). El grabador de salida puede liberar la memoria que se utiliza para MQCSP en esa etapa.

Cuando una instancia de canal de conexión del servidor comparte más de una conversación, el patrón de llamadas a la salida de seguridad se restringe en la segunda conversación y en las conversaciones posteriores.

Para la primera conversación, el patrón es el mismo que si la instancia de canal no comparte conversaciones. Para la segunda conversación y conversaciones posteriores, no se llama nunca a la salida de seguridad con MQXR_INIT, MQXR_INIT_SEC o MQXR_SEC_MSG. Se llama con MQXR_SEC_PARMS.

En una instancia de canal con conversaciones compartidas, se llama a MQXR_TERM sólo para la última conversación en ejecución.

Cada conversación tiene la oportunidad en la invocación MQXR_SEC_PARMS de la salida de alterar MQCD; en el extremo de conexión del servidor del canal esta característica puede ser útil para variar, por ejemplo, los valores MCAUserIdentifier o LongMCAUserIdPtr antes de realizar la conexión al gestor de colas.

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

Figura 4. Intercambio iniciado por la conexión con el cliente con acuerdo para conexión con el cliente utilizando parámetros de seguridad

Nota: Las aplicaciones de salida de seguridad creadas antes del release de IBM WebSphere MQ 7.1 pueden requerir actualización. Para obtener más información, consulte Programas de salida de seguridad de canal.

ALW Identificación de la llamada API en un programa de salidas de envío o recepción

Cuando utilice canales MQI para cliente, el byte 10 del almacenamiento intermedio del agente identifica la llamada API en uso cuando se ha llamado a una salida de envío o recepción. Esto resulta útil para identificar qué flujos de canales incluyen datos de usuario y pueden necesitar que se procesen como, por ejemplo, la firma digital o el cifrado.

En la tabla siguiente se muestran los datos que aparecen en el byte 10 del flujo de canales cuando se está procesando una llamada API.

Nota: No son los únicos valores de este byte. Hay otros valores **reservados**.

Tabla 8. Identificación de llamadas API

llamada de API	**Valor del byte 10 para la solicitud**	**Valor del byte 10 para la respuesta**
MQCONN "1" en la página 65,"2" en la página 65	X'81'	X'91'
MQDISC "1" en la página 65	X'82'	X'92'
MQOPEN "3" en la página 65	X'83'	X'93'
MQCLOSE	X'84'	X'94'
MQGET "4" en la página 65	X'85'	X'95'
MQPUT "4" en la página 65	X'86'	X'96'
Solicitud MQPUT1 "4" en la página 65	X'87'	X'97'
Solicitud MQSET	X'88'	X'98'
Solicitud MQINQ	X'89'	X'99'
Solicitud MQCMIT	X'8A'	X'9A'
Solicitud MQBACK	X'8B'	X'9B'
Solicitud MQSTAT	X'8D'	X'9D'
Solicitud MQSUB	X'8E'	X'9E'
Solicitud MQSUBRQ	X'8F'	X'9F'
Solicitud xa_start	X'A1'	X'B1'
Solicitud xa_end	X'A2'	X'B2'
Solicitud xa_open	X'A3'	X'B3'
Solicitud xa_close	X'A4'	X'B4'
Solicitud xa_prepare	X'A5'	X'B5'
Solicitud xa_commit	X'A6'	X'B6'
Solicitud xa_rollback	X'A7'	X'B7'
Solicitud xa_forget	X'A8'	X'B8'
Solicitud xa_recover	X'A9'	X'B9'
Solicitud xa_complete	X'AA'	X'BA'

Notas:

1. La conexión entre el cliente y el servidor la inicia la aplicación cliente mediante MQCONN. Además, para este determinado mandato, hay varios flujos de red. Lo mismo ocurre con MQDISC, que finaliza la conexión de red.
2. MQCONNX se trata de la misma forma que MQCONN en cuanto a la conexión cliente-servidor.
3. Si se abre una lista de distribución grande, puede que haya más de un flujo de redes por cada llamada MQOPEN para pasar todos los datos necesarios a SVRCONN MCA.
4. Los mensajes grandes pueden superar el tamaño de los segmentos de transmisión. Si se produjera esta situación, es posible que se generen muchos flujos de red a partir de una sola llamada API.

z/OS Connecting a client to a queue sharing group

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

About this task

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

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

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

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

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

Procedure

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

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

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

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

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

Results

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

Utilización de las variables de entorno de IBM MQ

Puede utilizar mandatos para visualizar los valores actuales o restablecer los valores de las variables de entorno de IBM MQ.

Acerca de esta tarea

Puede utilizar variables de entorno de las formas siguientes:

- Para establecer las variables en el perfil del sistema para efectuar un cambio permanente
- Para emitir un mandato desde la línea de mandatos para realizar un cambio para esta sesión solamente
- Para otorgar a una o más variables un valor determinado que dependa de la aplicación que se está ejecutando, añada mandatos a un archivo de script de mandatos utilizado por la aplicación.

Para cada variable de entorno, puede utilizar mandatos para visualizar el valor actual o para restablecer el valor de la variable de entorno. Estos mandatos están disponibles en todas las plataformas soportadas a menos que se indique lo contrario. El formato del mandato depende de la plataforma. Por ejemplo:

- Linux AIX En AIX and Linux:

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

- Windows En Windows:

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

- IBM i En IBM i:

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

- MQ Appliance Para IBM MQ Appliance, consulte Configuración de variables de entorno en IBM MQ Appliance en la documentación de IBM MQ Appliance .

Cuando sea aplicable, IBM MQ utiliza valores predeterminados para las variables de entorno que no ha establecido.

Nota: z/OS IBM MQ for z/OS no da soporte a las variables de entorno de IBM MQ. Si utiliza esta plataforma como su servidor, consulte Tabla de definiciones de canal de cliente para obtener información sobre cómo se genera la tabla de definiciones de canal de cliente en z/OS. Todavía puede utilizar variables de entorno de IBM MQ en la plataforma de cliente.

Procedimiento

- Windows

  En Windows, para cada variable de entorno, utilice los mandatos siguientes para visualizar el valor actual o para restablecer el valor de una variable:

  – Para eliminar el valor de una variable de entorno, utilice el mandato siguiente:

  ```
  SET MQSERVER=
  ```

  – Para mostrar el valor actual de una variable de entorno, utilice el mandato siguiente:

  ```
  SET MQSERVER
  ```

- Para mostrar todas las variables de entorno para la sesión, utilice el mandato siguiente:

  ```
  set
  ```

- Linux AIX
  En AIX and Linux, para cada variable de entorno, utilice los mandatos siguientes para visualizar el valor actual o para restablecer el valor de una variable:
  - Para eliminar el valor de una variable de entorno, utilice el mandato siguiente:

    ```
    unset MQSERVER
    ```

  - Para mostrar el valor actual de una variable de entorno, utilice el mandato siguiente:

    ```
    echo $MQSERVER
    ```

  - Para mostrar todas las variables de entorno para la sesión, utilice el mandato siguiente:

    ```
    set
    ```

Tareas relacionadas

Establecimiento de variables de entorno para IBM MQ classes for JMS/Jakarta Messaging

Variables de entorno aplicables a IBM MQ classes for Java

Definición de variables de entorno adicionales en el archivo service.env

“Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms” en la página 95

Puede cambiar el comportamiento de IBM MQ o de un gestor de colas individual para que se ajuste a las necesidades de la instalación editando la información en los archivos de configuración (`.ini`). También puede cambiar las opciones de configuración para IBM MQ MQI clients.

Referencia relacionada

El uso de variables de entorno en las propiedades MFT

Descripciones de variables de entorno

Descripciones de las variables de entorno de servidor y cliente que están pensadas para el uso del cliente.

Ejemplos de uso

- Linux AIX En sistemas AIX and Linux , utilice este formato: `export [environment variable]=value`.
- Windows En sistemas Windows , utilice este formato: `Set [environment variable]=value`.
- IBM i En sistemas IBM i , utilice este formato: `ADDENVVAR ENVVAR(environment variable) VALUE(xx)`.
- MQ Appliance Para IBM MQ Appliance, consulte Configuración de variables de entorno en IBM MQ Appliance en la documentación de IBM MQ Appliance .

Tabla 9. Resumen de variables de entorno

Nombre	Descripción	¿Está en el servidor?	¿Está en el cliente?
AMQ_ALLOWED_CIPHERS	Especifica una lista personalizada de CipherSpecs que están habilitadas para su uso con canales IBM MQ .	✓	

Tabla 9. Resumen de variables de entorno (continuación)			
Nombre	**Descripción**	**¿Está en el servidor?**	**¿Está en el cliente?**
AMQ_BAD_COMMS_DATA_FDCS	Hace que los archivos FFST se graben para los datos incorrectos, incluidos los formatos simples conocidos, si IBM MQ recibe datos que están en un formato incorrecto de un sistema principal a través de TCP/IP.	✓	
AMQ_CONVBCDICNEWLINE	Especifica cómo IBM MQ debe convertir un carácter NL EBCDIC en formato ASCII.	✓	
AMQ_DIAGNOSTIC_MSG_SEVERITY	Especifica si la gravedad del mensaje se debe añadir al número de mensaje cuando un proceso de IBM MQ escribe un mensaje en un registro de errores o en la consola.	✓	
AMQ_DISABLE_CLIENT_AMS	Para clientes Java , si desea conectarse a un gestor de colas desde una versión anterior del producto, inhabilita IBM MQ Advanced Message Security (AMS) en el cliente.		✓
AMQ_DMPMQCFG_QSGDISP_DEFAULT	Especifica qué tipos de definición incluir al consultar la disposición de un gestor de colas.	✓	
V 9.4.0 AMQ_IODELAY, AMQ_IODELAY_INMS y AMQ_IODELAY_FFST	Se utiliza para ajustar los diagnósticos y las temporizaciones cuando la entrada y salida para el registro del gestor de colas y el sistema de archivos de almacenamiento son lentos.	✓	
AMQ_LDAP_TRACE	Permite activar y desactivar el rastreo de cliente LDAP sin detener o iniciar también el gestor de colas.	✓	
Métrica AMQ_LICENSING_METRIC	Hace que el gestor de colas cargue datos relacionados con tipos de licencia de VPC mensuales en lugar del comportamiento predeterminado de cargar datos relacionados con licencias basadas en contenedor por horas.	✓	
Linux AIX UBICACIÓN_MQS_INI_AMQ_MQS	Especifica la ubicación que se utiliza para el archivo `mqs.ini` .	✓	
AMQ_NO_BAD_COMMS_DATA_FDCS	Suprime la generación de FFST al notificar mensajes de error en el flujo de comunicaciones inicial si los datos que IBM MQ recibe de un host a través de TCP/IP están en un formato incorrecto.	✓	
AMQ_NO_IPV6	Inhabilita el uso de IPv6 al intentar una conexión.	✓	✓

Tabla 9. Resumen de variables de entorno (continuación)			
Nombre	**Descripción**	**¿Está en el servidor?**	**¿Está en el cliente?**
AMQ_REVERSE_COMMIT_ORDER	Configura un gestor de colas para que en una transacción XA el cambio del gestor de colas IBM MQ se confirme después de que se haya completado la actualización de la base de datos correspondiente.	✓	
AMQ_SSL_ALLOW_DEFAULT_CERT	Permite que el certificado que una aplicación utiliza para conectarse a un gestor de colas sea un certificado predeterminado, siempre que haya un certificado predeterminado en el repositorio de claves y que el repositorio de claves no contenga un certificado personal con el prefijo `ibmwebspheremq`*`userid`*.	✓	
AMQ_SSL_LDAP_SERVER_VERSION	Especifica que los componentes criptográficos de IBM MQ utilizan LDAP v2 o LDAP v3 en los casos en los que los servidores CRL requieren que se utilice una versión específica del protocolo LDAP.	✓	
AIX AMQ_USE_ZLIBNX	En AIX, permite a los agentes de canal de mensajes (MCA) utilizar la biblioteca zlibNX acelerada por hardware para la compresión y descompresión de datos de mensaje cuando se utilizan las técnicas ZLIBFAST o ZLIBHIGH.	✓	
GMQ_MQ_LIB	Especifica la biblioteca de enlaces de cliente cuando las clases de automatización de IBM MQ para ActiveX (MQAX) se ejecutan en el cliente en lugar del servidor. **Atención:** Esta variable de entorno se elimina en 9.2.		
IBM i Linux AIX INICIO	En AIX, Linux y IBM i, especifica el nombre del directorio en el que se busca el archivo `mqclient.ini` .	✓	
Windows HOMEDRIVE y HOMEPATH	En Windows, se utiliza conjuntamente para especificar el nombre del directorio en el que se busca el archivo `mqclient.ini` .	✓	
LDAP_BASEDN	Entorno necesario para ejecutar un programa de ejemplo LDAP. Especifica el nombre distinguido base para la búsqueda de directorio.	✓	

Tabla 9. Resumen de variables de entorno (continuación)			
Nombre	**Descripción**	**¿Está en el servidor?**	**¿Está en el cliente?**
HOST DE LDAP	Opcional para ejecutar un programa de ejemplo LDAP. Especifica el nombre del host donde se ejecuta el servidor LDAP.	✓	
VERSIÓN_LDAP	Opcional para ejecutar un programa de ejemplo LDAP. Especifica la versión del protocolo LDAP que se va a utilizar.	✓	
MQ_CHANNEL_SUPPRESS_INTERVAL	Especifica el intervalo de tiempo, en segundos, durante el cual se deben suprimir los mensajes definidos con **MQ_CHANNEL_SUPPRESS_MSGS** para que no se graben en el registro de errores, junto con el número de veces que se permitirá que se produzca un mensaje durante el intervalo de tiempo especificado antes de que se suprima.	✓	
MQ_CHANNEL_SUPPRESS_MSGS	Suprime los mensajes de error de canal en el registro de errores.	✓	
MQ_CONNECT_TYPE	Se utiliza en combinación con el tipo de enlace especificado en el campo Opciones de la estructura MQCNO que se utiliza en una llamada MQCONNX.		✓
MQ_CROSS_QUEUE_ORDER_ALL	Especifica que el orden de colocación del mensaje se mantiene en una unidad de trabajo.	✓	
MQ_EPHEMERAL_PREFIX	Especifica la vía de acceso al directorio efímero del gestor de colas, en el que se conservan los datos del gestor de colas efímero, mientras se ejecuta el gestor de colas.	✓	
Windows VÍA_ACCESO_ARCHIVO_MQ	Se crea durante la instalación del paquete de tiempo de ejecución. Contiene los mismos datos que el registro de Windows .	✓	
MQ_JAVA_DATA_PATH	Especifica el directorio para la salida de registro y rastreo para IBM MQ classes for JMS y IBM MQ classes for Jakarta Messaging, y IBM MQ classes for Java.	✓	
MQ_JAVA_INSTALL_PATH	Especifica el directorio donde están instalados IBM MQ classes for JMS y IBM MQ classes for Jakarta Messaging, y IBM MQ classes for Java .	✓	
MQ_JAVA_LIB_PATH	Especifica el directorio donde se almacenan las bibliotecas IBM MQ classes for JMS y IBM MQ classes for Jakarta Messagingy IBM MQ classes for Java .	✓	

Tabla 9. Resumen de variables de entorno (continuación)			
Nombre	**Descripción**	**¿Está en el servidor?**	**¿Está en el cliente?**
"MQ_OVERRIDE_DATA_PATH" en la página 81	Altera temporalmente el directorio predeterminado de la vía de acceso de datos de IBM MQ .	✓	✓
AIX MQ_SET_NODELAYACK	En AIX, desactiva el acuse de recibo retardado de TCP.	✓	
"MQ_XX_ENCODE_CASE_ONE nombre_usuario" en la página 81	Permite que una instalación no registrada en Linux elija el nombre de un usuario sin nombre.	✓	
MQAPI_TRACE_LOGFILE	Define el prefijo del archivo especificado por el usuario en el que el programa de salida de API de ejemplo genera un rastreo de MQI.	✓	
ALW MQAPPLNAME	Si el nombre de aplicación todavía no se ha elegido, especifica el nombre que se debe utilizar para identificar la conexión con un gestor de colas.		✓
MQCCSID	Especifica el número de juego de caracteres codificado que debe utilizarse y altera temporalmente el valor CCSID con el que se ha configurado el servidor.		✓
MQCCDTURL	Proporciona la capacidad equivalente para establecer una combinación de las variables de entorno **MQCHLLIB** y **MQCHLTAB** .		✓
MQCERTLABL	Define la etiqueta de certificado de una definición de canal para que IBM MQ la utilice para localizar un certificado personal que se envía durante un reconocimiento TLS.		✓
MQCERTVPOL	Especifica el tipo de política de validación de certificados que se va a utilizar.		✓
MQCHLLIB	Especifica la vía de acceso del directorio al archivo que contiene la tabla de definición de canal de cliente (CCDT).		✓
MQCHLTAB	Especifica el nombre del archivo que contiene la tabla de definición de canal de cliente (CCDT).		✓
MQCLNTCF	Especifica la ubicación del archivo de configuración IBM MQ MQI client .		✓
MQDOTNET_TRACE_ON	Habilita el rastreo para clientes redistribuibles de IBM MQ .NET .		✓
MQIPADDRV	Especifica el protocolo IP que se tiene que utilizar en una conexión de canal.		✓

Tabla 9. Resumen de variables de entorno (continuación)			
Nombre	**Descripción**	**¿Está en el servidor?**	**¿Está en el cliente?**
MQKEYRPWD	Especifica la contraseña del repositorio de claves que contiene el certificado digital que pertenece al usuario.		✓
Linux MQLICENSE	En sistemas Linux , se utiliza para aceptar o ver una licencia de IBM MQ después de instalar el producto.	✓	✓
MQMAXERRORLOGSIZE	Especifica el tamaño del registro de errores del gestor de colas que se copia en la copia de seguridad.	✓	
Windows NOMBRE	Especifica el nombre de NetBIOS local que los procesos de IBM MQ pueden utilizar.	✓	✓
MQNOREMPOOL	Desactiva la agrupación de canales y hace que los canales se ejecuten como hebras del escucha.	✓	
MQPSE_TRACE_LOGFILE	Describe dónde deben grabarse los archivos de rastreo para el programa de ejemplo de salida de publicación.	✓	
MQS_AMSCRED_KEYFILE	Altera temporalmente o proporciona el archivo de claves inicial para utilizar en tiempo de ejecución de aplicaciones IBM MQ Advanced Message Security (AMS), o cuando está protegiendo un archivo de configuración de almacén de claves utilizando el mandato **runamscred** .		✓
MQS_DISABLE_ALL_INTERCEPT	Para los clientes C nativos, si desea conectarse a un gestor de colas desde una versión anterior del producto, inhabilita IBM MQ Advanced Message Security (AMS) en el cliente.		✓
MQS_IPC_HOST	Establece el nombre de host que se añade a la vía de acceso del directorio.	✓	
MQS_KEYSTORE_CONF	Especifica la ubicación del archivo de configuración del almacén de claves para IBM MQ Advanced Message Security (AMS), si el archivo no está en la ubicación predeterminada.		✓
MQS_MQI_KEYFILE	Especifica la ubicación de un archivo de claves inicial que contiene la clave inicial que se utilizará para las operaciones de protección de contraseña.		✓
MQS_SSLCRYP_KEYFILE	Especifica la vía de acceso completa y el nombre del archivo que contiene la clave inicial utilizada para cifrar la contraseña en la serie de configuración de hardware criptográfico PKCS #11 .		✓

Tabla 9. Resumen de variables de entorno (continuación)			
Nombre	**Descripción**	**¿Está en el servidor?**	**¿Está en el cliente?**
MQS_TRACE_OPTIONS	Para el rastreo selectivo de componentes en AIX, activa las funciones de rastreo de parámetros y de alto detalle de forma individual.	✓	✓
MQSERVER	Define un canal mínimo, especificando la ubicación del servidor IBM MQ y el método de comunicación que se va a utilizar.		✓
MQSNOAUT	Inhabilita el gestor de autorizaciones sobre objetos (OAM) e impide cualquier comprobación de seguridad, por ejemplo en un entorno de prueba.	✓	
MQSPREFIX	Como alternativa a cambiar el prefijo predeterminado, altera temporalmente **DefaultPrefix** para el mandato **crtmqm** .	✓	
ALW MQSSLCRYP	Contiene una serie de parámetros que puede utilizar para configurar el hardware criptográfico presente en el sistema.		✓
MQSSLFIPS	Especifica si sólo se van a utilizar algoritmos certificados por FIPS si se lleva a cabo el cifrado en IBM MQ.		✓
MQSSLKEYR	Especifica la ubicación del repositorio de claves que contiene el certificado digital que pertenece al usuario.		Cliente ✓ y herramienta **mqcertck**
MQSSLPROXY	Especifica el nombre de host y el número de puerto del servidor proxy HTTP que utilizará IBM Global Security Kit (GSKit) para las comprobaciones de OCSP.		✓
MQSSLRESET	Especifica el número de bytes no cifrados enviados y recibidos en un canal TLS antes de que se renegocie la clave secreta TLS.		✓
ALW MQSUITEB	Especifica si se va a utilizar la criptografía compatible con Suite B. En la instancia en la que se utiliza la criptografía Suite B,		✓
MQTCPTIMEOUT	especifica cuánto tiempo espera IBM MQ una llamada de conexión TCP.	✓	✓
ODQ_MSG	Establece el nombre del archivo que contiene los mensajes de error e información, si utiliza un manejador de cola de mensajes no entregados que es diferente de **runmqdlq**.	✓	

Tabla 9. Resumen de variables de entorno (continuación)			
Nombre	**Descripción**	**¿Está en el servidor?**	**¿Está en el cliente?**
ODQ_TRACE	Habilita el rastreo si utiliza un manejador de cola de mensajes no entregados que es diferente de **runmqdlq**.	✓	
WCF_TRACE_ON	Habilita el rastreo para el canal personalizado WCF.		✓
WMQSOAP_HOME	Se utiliza al completar pasos de configuración adicionales después de que el entorno de alojamiento de servicios .NET SOAP sobre JMS se haya instalado y configurado correctamente en IBM MQ.		✓
XMS_TRACE_ON, XMS_TRACE_FILE_PATH, XMS_TRACE_FORMAT y XMS_TRACE_SPECIFICATION	Se utiliza para habilitar y configurar el rastreo de XMS .		✓

AMQ_ALLOWED_CIPHERS

Multi

Puede utilizar la variable de entorno **AMQ_ALLOWED_CIPHERS** para especificar una lista personalizada de CipherSpecs que están habilitadas para su uso con canales IBM MQ en Multiplatforms. La variable de entorno toma los mismos valores que el atributo de stanza SSL de **AllowedCipherSpecs** del archivo `.ini` :

- Un solo nombre de CipherSpec o
- Una lista separada por comas de nombres de IBM MQ CipherSpec para volver a habilitar, o
- El valor especial de ALL, que representa todas las CipherSpecs (no se recomienda).

Nota: No se recomienda habilitar **ALL** (todas las) CipherSpecs ya que esto habilitará los protocolos SSL 3.0 y TLS 1.0 y un gran número de algoritmos criptográficos débiles.

Para obtener más información, consulte Proporcionar una lista personalizada de CipherSpecs habilitadas en Multiplatforms en el orden CipherSpec en el reconocimiento TLS.

AMQ_BAD_COMMS_DATA_FDCS

La variable de entorno **AMQ_BAD_COMMS_DATA_FDCS** es efectiva cuando se establece en cualquier valor.

Si los datos que IBM MQ recibe de un host a través de TCP/IP están en un formato incorrecto, por ejemplo, porque un cliente de red se ha conectado a un puerto de escucha de IBM MQ y ha intentado comunicarse con un protocolo de aplicación no soportado, el gestor de colas graba un mensaje de error AMQ9207E en los registros de errores del gestor de colas. Los escuchas de IBM MQ dan soporte a las conexiones TCP/IP de los agentes de canal de mensajes (MCA) del gestor de colas y de las aplicaciones cliente MQI, JMS y XMS .

Nota: Los escuchas de IBM MQ no dan soporte al protocolo de aplicación utilizado por los clientes AMQP y MQTT; en su lugar, estos clientes deben conectarse a los puertos de red configurados en el canal AMQP o servicio de telemetría MQXR aplicable.

Es posible que también se grabe un registro de captura de datos de anomalía (FDC) que contenga los datos no válidos que IBM MQ ha recibido. Sin embargo, no se genera un archivo FFST si este es el principio de una conversación con el lado remoto y el formato es un formato simple conocido como una solicitud GET de un navegador web HTTP. Si desea alterar temporalmente esto para que los archivos FFST se graben para los datos incorrectos incluidos los formatos simples conocidos, puede establecer la variable de entorno **AMQ_BAD_COMMS_DATA_FDCS** en cualquier valor (por ejemplo, TRUE) y reiniciar el gestor de colas.

AMQ_CONVBCDICNEWLINE

Puede utilizar la variable de entorno **AMQ_CONVEBCDICNEWLINE** para especificar cómo IBM MQ va a convertir un carácter NL EBCDIC en formato ASCII. La variable de entorno toma los mismos valores que el atributo **ConvEBCDICNewline** del `mqs.ini`, es decir, `NL_TO_LF`, `TABLE`o `ISO` (consulte Stanza de todos los gestores de colas del archivo mqs.ini). Por ejemplo, puede utilizar la variable de entorno **AMQ_CONVEBCDICNEWLINE** en lugar del atributo de stanza **ConvEBCDICNewline** para proporcionar la funcionalidad **ConvEBCDICNewline** en el lado del cliente en situaciones en las que no se puede utilizar el archivo `mqs.ini` . Si se establecen tanto el atributo de stanza como la variable de entorno, el atributo de stanza tiene prioridad.

Para obtener más información, consulte Conversión de datos entre juegos de caracteres codificados .

AMQ_DIAGNOSTIC_MSG_SEVERITY

Si la variable de entorno **AMQ_DIAGNOSTIC_MSG_SEVERITY** se establece en 1 para un proceso IBM MQ , esto hace que la gravedad del mensaje se añada al número de mensaje como un único carácter alfabético en mayúsculas cuando el proceso IBM MQ escribe un mensaje en un registro de errores o en la consola.

El comportamiento que **AMQ_DIAGNOSTIC_MSG_SEVERITY** habilita está establecido de forma predeterminada. Puede desactivar este comportamiento estableciendo la variable de entorno en `0`.

Si desea más información, consulte Utilización de registros de errores.

AMQ_DISABLE_CLIENT_AMS

Puede utilizar la variable de entorno **AMQ_DISABLE_CLIENT_AMS** para inhabilitar IBM MQ Advanced Message Security (AMS) en el cliente si se notifica un error `2085 (MQRC_UNKNOWN_OBJECT_NAME)` cuando intenta conectarse a un gestor de colas desde una versión anterior del producto y está utilizando uno de los clientes siguientes:

- Un Java runtime environment (JRE) que no sea IBM Java runtime environment (JRE)
- Un cliente IBM MQ IBM MQ classes for JMS o IBM MQ classes for Java .

Nota: No puede utilizar la variable de entorno **AMQ_DISABLE_CLIENT_AMS** para clientes C. En su lugar, debe utilizar la variable de entorno **MQS_DISABLE_ALL_INTERCEPT** .

Para obtener más información, consulte Inhabilitación de Advanced Message Security en el cliente.

AMQ_DMPMQCFG_QSGDISP_DEFAULT

Las consultas sobre la disposición de un gestor de colas que utiliza el mandato **dmpmqcfg** sólo consultan las definiciones QSGDISP (QMGR) de forma predeterminada. Puede consultar definiciones adicionales utilizando la variable de entorno **AMQ_DMPMQCFG_QSGDISP_DEFAULT** , que se puede establecer en uno de los valores siguientes:

DIRECTO
Incluir sólo objetos definidos con QSGDISP (QMGR) o QSGDISP (COPY).

TODOS
Incluir objetos definidos con QSGDISP (QMGR) y QSGDISP (COPY). Si el gestor de colas es miembro de un grupo de compartición de colas, también se incluyen QSGDISP (GROUP) y QSGDISP (SHARED).

COPY
Incluir sólo, objetos definidos con QSGDISP (COPY)

GRUPO
Incluir sólo objetos definidos con QSGDISP (GROUP); el gestor de colas de destino debe ser miembro de un grupo de compartición de colas.

QMGR
Incluir sólo objetos definidos con QSGDISP (QMGR). Este es el comportamiento predeterminado si utiliza esta variable de entorno, para que coincida con el comportamiento existente de **dmpmqcfg**.

PRIVATE
Incluir sólo objetos definidos con QSGDISP (QMGR) o QSGDISP (COPY).

SHARED
Incluir sólo objetos definidos con QSGDISP (SHARED).

AMQ_IODELAY, AMQ_IODELAY_INMS y AMQ_IODELAY_FFST

IBM MQ detecta cuando las operaciones de lectura y grabación o de entrada y salida de registro tardan más de lo previsto. Esto puede deberse a problemas con el sistema operativo o el sistema de almacenamiento y puede afectar al rendimiento del gestor de colas. A partir de IBM MQ 9.4.0, puede utilizar las variables de entorno de **AMQ_IODELAY** para ajustar los diagnósticos y las temporizaciones cuando la entrada y salida para el sistema de archivos de registro y almacenamiento del gestor de colas es lenta. Si ve el mensaje AMQ6729W `Registrar umbral de operación de E/S excedido` en el registro de errores del gestor de colas, investigue la causa y realice los ajustes correspondientes. Utilice las variables tal como se muestra en los ejemplos siguientes:

AMQ_IODELAY
Tiempo de umbral en segundos, el valor predeterminado es 1 segundo. Si una operación de E/S tarda más de este umbral, el mensaje de error `AMQ6729W` se notifica en los archivos de registro de IBM MQ . El mensaje de aviso se repite como máximo cada 10 segundos si los retardos continúan. Puede aumentarlo para suprimir errores o disminuir para investigar problemas de rendimiento específicos. Por ejemplo,

```
export AMQ_IODELAY=200000
```

AMQ_IODELAY_INMS
Cambie la medida de tiempo a microsegundos en lugar de segundos. Utilice esta opción para establecer un umbral inferior antes de obtener el mensaje `AMQ6729` en el registro del gestor de colas.

```
export AMQ_IODELAY_INMS=YES
```

AMQ_IODELAY_FFST
Además del mensaje de aviso en el registro de errores, se genera un archivo FFST que contiene información de diagnóstico siempre que se supera el umbral.

```
export AMQ_IODELAY_FFST=YES
```

Al iniciar el gestor de colas como en este ejemplo, hace que se escriba un archivo FDC o FFST si una operación de entrada/salida tarda más de 200000 microsegundos (0.2s), que sigue siendo un umbral relativamente generoso.

Para obtener más información, consulte Comportamiento de comprobación de estado del gestor de colas.

AMQ_LDAP_TRACE

Si la variable de entorno **AMQ_LDAP_TRACE** se establece en un valor no nulo, es posible activar y desactivar el rastreo de cliente LDAP sin detener o iniciar también el gestor de colas.

Para obtener más información, consulte Habilitación del rastreo dinámico del código de la biblioteca del cliente LDAP.

MÉTRICA_LICENCIA_AMQ

Multi

El establecimiento de la variable de entorno **AMQ_LICENSING_METRIC=VPCMonthlyPeak** hace que el gestor de colas cargue datos relacionados con los tipos de licencia de VPC mensuales, en lugar del comportamiento predeterminado de la carga de datos relacionados con las licencias basadas en contenedor por horas.

Para obtener más información sobre cómo configurar IBM MQ para su uso con el servicio de calibración de IBM Cloud Private , consulte Servicio de calibración deIBM Cloud Private en la documentación de IBM Cloud Private .

UBICACIÓN_UBICACIÓN_INICIO_MQS_AMQ

Linux AIX

En sistemas AIX and Linux , puede modificar la ubicación que se utiliza para el archivo `mqs.ini` estableciendo la ubicación del archivo `mqs.ini` en la variable de entorno **AMQ_MQS_INI_LOCATION** . Esta variable de entorno debe establecerse en el nivel del sistema.

Para obtener más información sobre el archivo `mqs.ini` , incluidas las ubicaciones de directorio, consulte el archivo de configuración IBM MQ , mqs.ini.

AMQ_NO_BAD_COMMS_DATA_FDCS

La variable de entorno **AMQ_NO_BAD_COMMS_DATA_FDCS** es efectiva cuando se establece en cualquier valor.

Si IBM MQ no reconoce la transmisión de datos inicial al intentar conectar un cliente noIBM MQ a un escucha TCP/IP de IBM MQ , esto hace que el gestor de colas escriba un mensaje de error AMQ9207E en los registros de errores del gestor de colas. También se graba un registro de captura de datos de anomalía (FDC). Puede suprimir la generación de estos archivos de diagnóstico con la variable de entorno **AMQ_NO_BAD_COMMS_DATA_FDCS** . Cuando **AMQ_NO_BAD_COMMS_DATA_FDCS** se establece en cualquier valor (por ejemplo, TRUE), esto indica a IBM MQ que no genere FFST al notificar AMQ9207E mensajes de error en el flujo de comunicaciones inicial. Para que sea efectiva, la variable de entorno debe establecerse antes de iniciar el gestor de colas y los procesos de escucha.

El FDC se sigue generando en el caso en el que un cliente envía flujos de protocolo IBM MQ válidos al gestor de colas y, a continuación, envía datos no válidos, ya que esto es indicativo de un problema de cliente que justifica una investigación adicional.

Nota: La captura de FFST al informar de mensajes de error de AMQ9207E en flujos de comunicaciones iniciales se suprime de forma predeterminada.

AMQ_NO_IPV6

La variable de entorno **AMQ_NO_IPV6** es efectiva cuando se establece en cualquier valor. Cuando se establece esta variable de entorno, inhabilita el uso de IPv6 al intentar una conexión.

AMQ_REVERSE_COMMIT_ORDER

La variable de entorno **AMQ_REVERSE_COMMIT_ORDER** configura un gestor de colas para que en una transacción XA el cambio del gestor de colas IBM MQ se confirme después de que se haya completado la actualización de la base de datos correspondiente. Las aplicaciones que leen mensajes de las colas sólo ven un mensaje después de que se haya completado la actualización de la base de datos correspondiente.

Nota: No establezca **AMQ_REVERSE_COMMIT_ORDER** sin leer y comprender el escenario que se describe en Nivel de aislamiento.

AMQ_SSL_ALLOW_DEFAULT_CERT

Cuando la variable de entorno **AMQ_SSL_ALLOW_DEFAULT_CERT** no está establecida, una aplicación puede conectarse a un gestor de colas con un certificado personal en el almacén de claves del cliente sólo cuando el certificado incluye el nombre de etiqueta de `ibmwebspheremq`*`userid`*. Cuando se establece la variable de entorno **AMQ_SSL_ALLOW_DEFAULT_CERT** , el certificado no requiere el nombre de etiqueta de `ibmwebspheremq`*`userid`*. Es decir, el certificado que se utiliza para conectarse a un gestor de colas puede ser un certificado predeterminado, siempre que esté presente un certificado predeterminado en el repositorio de claves y el repositorio de claves no contenga un certificado personal con el prefijo `ibmwebspheremq`*`userid`*.

Un valor de 1 habilita el uso de un certificado predeterminado.

En lugar de utilizar la variable de entorno **AMQ_SSL_ALLOW_DEFAULT_CERT** , una aplicación puede utilizar el valor **`CertificateLabel`** de la stanza SSL en el archivo `mqclient.ini` . Para obtener más información, consulte Etiquetas de certificado digital, que comprenden los requisitos y stanza SSL del archivo de configuración del cliente.

AMQ_SSL_LDAP_SERVER_VERSION

La variable de entorno **AMQ_SSL_LDAP_SERVER_VERSION** se puede utilizar para asegurarse de que los componentes criptográficos de IBM MQ utilizan LDAP v2 o LDAP v3 en los casos en los que los servidores CRL requieren que se utilice una versión específica del protocolo LDAP.

Establezca la variable de entorno en el valor adecuado en el entorno que se utiliza para iniciar el gestor de colas o canal:

- Para solicitar que se utilice LDAP v2 , establezca `AMQ_SSL_LDAP_SERVER_VERSION=2`.
- Para solicitar que se utilice LDAP v3 , establezca `AMQ_SSL_LDAP_SERVER_VERSION=3`.

Esta variable de entorno no afecta a las conexiones LDAP establecidas por el gestor de colas de IBM MQ para la autenticación de usuarios o la autorización de usuarios.

AMQ_USE_ZLIBNX

AIX

En AIX, la variable de entorno **AMQ_USE_ZLIBNX** se puede utilizar para permitir que los agentes de canal de mensajes (MCA) utilicen la biblioteca zlibNX acelerada por hardware para la compresión y descompresión de datos de mensaje cuando se utilizan las técnicas ZLIBFAST o ZLIBHIGH.

Consejo: Es más probable que los mensajes altamente comprimibles de más de 2 KB de tamaño se beneficien de optar por utilizar la biblioteca zlibNX , reduciendo el uso de CPU.

La biblioteca zlibNX está disponible en IBM AIX 7.2 con Technology Level 4 Expansion Pack y posteriores. Si la variable de entorno está establecida y la biblioteca zlibNX (`/usr/opt/zlibNX/lib/libz.a`) no está instalada, los agentes de canal de mensajes utilizarán la biblioteca zlib estándar proporcionada en la instalación de IBM MQ for AIX .

INICIO

IBM i Linux AIX

En AIX, Linux y IBM i, la variable de entorno **HOME** especifica el nombre del directorio en el que se busca el archivo `mqclient.ini` . Este archivo contiene información de configuración que utiliza IBM MQ MQI clients.

Para obtener más información, consulte Archivo de configuración del cliente MQI deIBM MQ , mqclient.ini y Ubicación del archivo de configuración del cliente.

HOMEDRIVE y HOMEPATH

Windows

Para utilizar, se deben establecer las variables de entorno **HOMEDRIVE** y **HOMEPATH** . Se utilizan en sistemas Windows para especificar el nombre del directorio en el que se busca el archivo `mqclient.ini` . Este archivo contiene información de configuración que utiliza IBM MQ MQI clients.

Para obtener más información, consulte Archivo de configuración del cliente MQI deIBM MQ , mqclient.ini y Ubicación del archivo de configuración del cliente.

LDAP_BASEDN

LDAP_BASEDN es la variable de entorno necesaria para ejecutar un programa de ejemplo LDAP. Especifica el nombre distinguido base para la búsqueda de directorio.

HOST DE LDAP

LDAP_HOST es una variable de entorno opcional para ejecutar un programa de ejemplo LDAP. Especifica el nombre del host en el que se está ejecutando el servidor LDAP; toma el valor predeterminado del host local si no se especifica.

VERSIÓN_LDAP

LDAP_VERSION es una variable de entorno opcional para ejecutar un programa de ejemplo LDAP. Especifica la versión del protocolo LDAP que se va a utilizar y puede ser 2 o 3. La mayoría de los servidores LDAP ahora dan soporte a la versión 3 del protocolo; todos ellos dan soporte a la versión 2 anterior. Este ejemplo funciona igual de bien con cualquiera de las versiones del protocolo y, si no se especifica, toma el valor predeterminado de la versión 2.

MQ_CHANNEL_SUPPRESS_INTERVAL

La variable de entorno **MQ_CHANNEL_SUPPRESS_INTERVAL** especifica el intervalo de tiempo, en segundos, durante el cual los mensajes definidos con **MQ_CHANNEL_SUPPRESS_MSGS** deben suprimirse de la grabación en el registro de errores, junto con el número de veces que se permitirá que se produzca un mensaje durante el intervalo de tiempo especificado antes de que se suprima. El valor predeterminado es 60,5, lo que significa que cualquier otra aparición de un mensaje determinado se suprime después de las cinco primeras apariciones de dicho mensaje en un intervalo de 60 segundos. Para obtener más información, consulte Supresión de mensajes de error de canal de registros de errores en Multiplatforms.

La variable de entorno **MQ_CHANNEL_SUPPRESS_INTERVAL** es comparable a SuppressInterval en el archivo `qm.ini` .

MQ_CHANNEL_SUPPRESS_MSGS

La variable de entorno **MQ_CHANNEL_SUPPRESS_MSGS** suprime los mensajes de error de canal en el registro de errores. Puede especificar una lista de mensajes que se suprimen. **MQ_CHANNEL_SUPPRESS_MSGS** se utiliza junto con **MQ_CHANNEL_SUPPRESS_INTERVAL**, que especifica el número de veces que aparece cada mensaje antes de que se suprima y el periodo de tiempo durante el que se suprimen los mensajes. Para obtener más información, consulte Supresión de mensajes de error de canal de registros de errores en Multiplatforms.

La variable de entorno **MQ_CHANNEL_SUPPRESS_MSGS** es comparable a SuppressMessage en el archivo `qm.ini` , excepto que puede suprimir cualquier mensaje de canal utilizando la variable de entorno, mientras que existe una lista restrictiva para el método `qm.ini` .

MQ_CONNECT_TYPE

Multi

En Multiplatforms, puede utilizar la variable de entorno **MQ_CONNECT_TYPE** en combinación con el tipo de enlace especificado en el campo Opciones de la estructura MQCNO que se utiliza en una llamada MQCONNX. **MQ_CONNECT_TYPE** sólo tiene efecto para los enlaces STANDARD. Para otros enlaces, se ignora **MQ_CONNECT_TYPE** .

Para obtener más información, consulte Utilización de opciones de llamada MQCONNX con MQ_CONNECT_TYPE.

MQ_CROSS_QUEUE_ORDER_ALL

Cuando establece la variable de entorno **MQ_CROSS_QUEUE_ORDER_ALL** en un valor distinto de cero, el orden de colocación del mensaje se mantiene en una unidad de trabajo. Esto significa que, si los mensajes de una unidad de trabajo (UoW) se colocan en varias colas (por ejemplo, Q1, entonces Q2), cuando se emite un MQCMIT, los mensajes se entregan y pasan a estar disponibles en el mismo orden de cola en el que estaban PUT.

En un entorno de gestor de varias colas, **MQ_CROSS_QUEUE_ORDER_ALL** debe existir y tener un valor no vacío en el lado emisor y receptor antes de que se inicie cada gestor de colas.

PREFIJO_PREFIJO_EFEMÉRIDE

La variable de entorno **MQ_EPHEMERAL_PREFIX** especifica la vía de acceso al directorio efímero del gestor de colas, dentro del cual se conservan los datos del gestor de colas efímero, mientras se ejecuta el gestor de colas.

Como alternativa a cambiar el prefijo efímero cambiando el atributo **EphemeralPrefix** en el atributo **DefaultEphemeralPrefix** de la stanza AllQueueManagers del archivo `mqs.ini` , puede utilizar la variable de entorno **MQ_EPHEMERAL_PREFIX** para alterar temporalmente el **EphemeralPrefix** para el mandato **crtmqm** . Para obtener más información, consulte Directorio efímero configurable.

MQ_FILE_PATH

Windows

La variable de entorno **MQ_FILE_PATH** se configura durante la instalación del paquete de tiempo de ejecución en la plataforma Windows . Esta variable de entorno contiene los mismos datos que la clave siguiente en el registro de Windows :

```
HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphere MQ\Installation\InstallationName\FilePath
```

Para obtener más información, consulte setmqenv (set IBM MQ environment) y crtmqenv (create IBM MQ environment).

MQ_JAVA_DATA_PATH

La variable de entorno **MQ_JAVA_DATA_PATH** especifica el directorio para la salida de registro y rastreo para IBM MQ classes for JMS y IBM MQ classes for Jakarta Messaging y IBM MQ classes for Java. Lo utilizan los scripts proporcionados con IBM MQ classes for JMS y IBM MQ classes for Jakarta Messaging y IBM MQ classes for Java.

Para obtener más información, consulte Establecimiento de variables de entorno para clases IBM MQ para JMS/Jakarta Messaging y Variables de entorno relevantes para IBM MQ classes for Java.

MQ_JAVA_INSTALL_PATH

La variable de entorno **MQ_JAVA_INSTALL_PATH** especifica el directorio donde están instalados IBM MQ classes for JMS y IBM MQ classes for Jakarta Messaging tal como se muestra en Qué está instalado para IBM MQ classes for JMS, y IBM MQ classes for Java tal como se muestra en los directorios de instalación deIBM MQ classes for Java .

Para obtener más información, consulte Establecimiento de variables de entorno para clases IBM MQ para JMS/Jakarta Messaging y Variables de entorno relevantes para IBM MQ classes for Java.

MQ_JAVA_LIB_PATH

La variable de entorno **MQ_JAVA_LIB_PATH** especifica el directorio donde se almacenan las bibliotecas IBM MQ classes for JMS y IBM MQ classes for Jakarta Messagingy IBM MQ classes for Java . Algunos scripts, por ejemplo, IVTRun, que se proporcionan con IBM MQ classes for JMS y IBM MQ classes for Jakarta Messaging o IBM MQ classes for Java utilizan esta variable de entorno.

Para obtener más información, consulte Establecimiento de variables de entorno para clases IBM MQ para JMS/Jakarta Messaging y Variables de entorno relevantes para IBM MQ classes for Java.

MQ_OVERRIDE_DATA_PATH

Puede utilizar la variable de entorno **MQ_OVERRIDE_DATA_PATH** para cambiar el directorio predeterminado de la vía de acceso de datos IBM MQ .

MQ_SET_NODELAYACK

AIX

La variable de entorno **MQ_SET_NODELAYACK** desactiva el acuse de recibo retardado de TCP en AIX.

Cuando establece esta variable de entorno, el valor desactiva el acuse de recibo retardado de TCP llamando a la llamada setsockopt del sistema operativo con la opción TCP_NODELAYACK . Solo AIX da soporte a esta función, por lo que la variable de entorno **MQ_SET_NODELAYACK** sólo tiene un efecto en AIX.

MQ_XX_ENCODE_CASE_ONE nombre_usuario

Linux

Puede utilizar la variable de entorno **MQ_USER_NAME** para permitir que una instalación no registrada en Linux elija el nombre de un usuario sin nombre. Esto es necesario, por ejemplo, para utilizar jerarquías de publicación/suscripción en OpenShift.

El valor de **MQ_USER_NAME** no debe coincidir con un usuario preexistente en el sistema y debe ser menor o igual que 12 bytes.

MQAPI_TRACE_LOGFILE

El programa de salida de API de ejemplo genera un rastreo MQI para un archivo especificado por el usuario con un prefijo definido en la variable de entorno **MQAPI_TRACE_LOGFILE** .

Para obtener más información, consulte El programa de ejemplo de salida de API.

MQApplName

ALW

Si el nombre de aplicación todavía no se ha elegido, puede utilizar la variable de entorno **MQAPPLNAME** como el nombre que se utilizará para identificar la conexión con un gestor de colas. Sólo se utilizan los primeros 28 caracteres, y no deben ser todos blancos o nulos.

Para obtener más información, consulte Utilización del nombre de aplicación en lenguajes de programación soportados.

MQCCSID

La variable de entorno **MQCCSID** especifica el número de juego de caracteres codificado que se va a utilizar y altera temporalmente el valor CCSID con el que se ha configurado el servidor. **MQCCSID** se puede

utilizar para alterar temporalmente el CCSID nativo de una aplicación y especificar el número de juego de caracteres codificado que se va a utilizar, por ejemplo, si el CCSID nativo es un CCSID no soportado o no es el CCSID necesario.

Para establecer **MQCCSID**, utilice uno de los mandatos siguientes:

- Linux AIX En AIX and Linux:

```
export MQCCSID=number
```

- Windows En Windows:

```
SET MQCCSID=number
```

- IBM i En IBM i:

```
ADDENVVAR ENVVAR(MQCCSID) VALUE(number)
```

Para obtener más información, consulte Selección de CCSID de cliente o servidor.

MQCCDTURL

La variable de entorno **MQCCDTURL** proporciona la capacidad equivalente para establecer una combinación de las variables de entorno **MQCHLLIB** y **MQCHLTAB** . Le permite proporcionar un archivo, ftp o URL http como un único valor a partir del cual se puede obtener una tabla de definición de canal de cliente para programas nativos que se conectan como clientes, es decir, aplicaciones C, COBOL o C++.

Nota: El uso de variables de entorno para proporcionar el URL no tiene ningún efecto para las aplicaciones Java, JMS o .NET gestionadas.

IBM MQ permite recuperar una tabla de definición de canal de cliente a partir de un archivo, FTP o un URL HTTP. Sin embargo, **MQCCDTURL** sólo acepta un valor de URL. No acepta el formato de directorio del sistema de archivos local existente.

Para utilizar **MQCCDTURL** en lugar de **MQCHLLIB** y **MQCHLTAB** con un archivo local, puede utilizar un protocolo `'file://'` . Por lo tanto, como se muestra en este ejemplo para AIX y Linux:

```
export MQCCDTURL=file:///var/mqm/qmgrs/QMGR/@ipcc/MYCHL.TAB
```

es equivalente a:

```
export MQCHLLIB=/var/mqm/qmgrs/QMGR/@ipcc
export MQCHLTAB=MYCHL.TAB
```

También puede especificar un archivo JSON tal como se muestra en este ejemplo para Windows:

```
set MQCCDTURL=file:/c:/mq-channels/CCDT-QMGR1.json
```

es equivalente a:

```
set MQCHLLIB=C:\mq-channels
set MQCHLTAB=CCDT-QMGR1.json
```

Para obtener más información, consulte Acceso de URL a la CCDT.

MQCERTLABL

La variable de entorno **MQCERTLABL** define la etiqueta de certificado de una definición de canal para que IBM MQ la utilice para localizar un certificado personal que se envía durante un reconocimiento TLS.

Para obtener más información, consulte Etiquetas de certificado digital, que comprenden los requisitos.

MQCERTVPOL

La variable de entorno **MQCERTVPOL** especifica el tipo de política de validación de certificados que se va a utilizar. Esta variable de entorno altera temporalmente el atributo **CertificateValPolicy** en la stanza SSL del archivo de configuración del cliente.

MQCERTVPOL se puede establecer en uno de dos valores:

CUALQUIERA
Utilice cualquier política de validación de certificados soportada por la biblioteca de sockets seguros subyacente. Este valor es el predeterminado.

RFC5280
Utilice sólo la validación de certificados que cumpla con el estándar RFC 5280.

Para establecer **MQCERTVPOL**, utilice uno de estos mandatos:

- Linux AIX Para sistemas AIX and Linux:

```
export MQCERTVPOL= value
```

- Windows Para sistemas Windows:

```
SET MQCERTVPOL= value
```

- IBM i Para sistemas IBM i:

```
ADDENVVAR ENVVAR(MQCERTVPOL) VALUE(value)
```

Para obtener más información, consulte Políticas de validación de certificados en IBM MQ y Configuración de políticas de validación de certificados en IBM MQ.

MQCHLLIB

La variable de entorno **MQCHLLIB** especifica la vía de acceso del directorio al archivo que contiene la tabla de definición de canal de cliente (CCDT). El archivo se crea en el servidor, pero se puede copiar en la estación de trabajo de IBM MQ MQI client.

Para establecer **MQCHLLIB**, utilice uno de estos mandatos:

- Windows En Windows:

```
SET MQCHLLIB=pathname
```

  Por ejemplo:

```
SET MQCHLLIB=C:\wmqtest
```

- Linux AIX Para sistemas AIX and Linux:

```
export MQCHLLIB=pathname
```

- IBM i Para IBM i:

```
ADDENVVAR ENVVAR(MQCHLLIB) VALUE(pathname)
```

Si **MQCHLLIB** no está establecido, la vía de acceso del cliente toma el valor predeterminado siguiente:

- Linux AIX En AIX and Linux: `/var/mqm/`
- Windows En Windows: *MQ_INSTALLATION_PATH*
- IBM i En IBM i: `/QIBM/UserData/mqm/`

Para mandatos **crtmqm** y **strmqm**, la vía de acceso tendrá como valor predeterminado uno de los dos conjuntos de vías de acceso. Si se establece *datapath* , la vía de acceso toma como valor predeterminado uno de los primeros conjuntos. Si no se ha establecido *datapath* , la vía de acceso se establece de forma predeterminada en uno de los segundos conjuntos.

- Linux AIX En AIX and Linux: *datapath*`/@ipcc`
- Windows En Windows: *datapath*`\@ipcc`
- IBM i En IBM i: *datapath*`/&ipcc`

O:

- Linux AIX En AIX and Linux: `/prefix/qmgrs/`*qmgrname*`/@ipcc`
- Windows En Windows: *MQ_INSTALLATION_PATH*`\data\qmgrs\`*qmgrname*`\@ipcc`
- IBM i En IBM i: `/`*prefix*`/qmgrs/`*qmgrname*`/&ipcc`

donde:

- *MQ_INSTALLATION_PATH* representa el directorio de alto nivel en el que está instalado IBM MQ.
- Si está presente, *datapath* es el valor de `DataPath` definido en la stanza del gestor de colas.
- *prefix* es el valor de Prefijo definido en la stanza del gestor de colas. El prefijo suele ser uno de los valores siguientes:
  - Linux AIX `/var/mqm` en sistemas AIX and Linux.
  - IBM i `/QIBM/UserData/mqm/` en IBM i.
- *qmgrname* es el valor del atributo `Directory` definido en la stanza del gestor de colas. El valor puede ser diferente del nombre del gestor de colas real. El valor puede haberse alterado para sustituir caracteres especiales.
- El lugar donde se define la stanza del gestor de colas depende de la plataforma:
  - IBM i Linux AIX En el archivo `mqs.ini` en IBM i, AIX and Linux.
  - Windows En el registro, en Windows.

Notas:

1. z/OS Si está utilizando IBM MQ for z/OS como el servidor, el archivo debe mantenerse en la estación de trabajo de cliente de IBM MQ.
2. Si se ha establecido, `MQCHLLIB` altera temporalmente la vía de acceso utilizada para localizar la CCDT.
3. `MQCHLLIB` puede contener un URL que trabaja conjuntamente con la variable de entorno MQCHLTAB (consulte “Acceso de URL a la tabla de definición de canal de cliente” en la página 55).
4. Las variables de entorno, como **MQCHLLIB**, pueden ser del ámbito de un proceso o de un trabajo o de todo el sistema, según la plataforma.
5. Si establece **MQCHLLIB** en todo el sistema en un servidor, establece la misma vía de acceso en el archivo CCDT para todos los gestores de colas del servidor. Si no establece la variable de entorno **MQCHLLIB**, la vía de acceso es diferente para cada gestor de colas. Los gestores de colas leen el valor de **MQCHLLIB**, si está establecido, en el mandato **crtmqm** o **strmqm**.

6. Si crea varios gestores de colas en un servidor, la distinción es importante, por la siguiente razón. Si establece **MQCHLLIB** en todo el sistema, cada gestor de colas actualiza el mismo archivo CCDT. El archivo las definiciones de conexión con el cliente de todos los gestores de colas en el servidor. Si existe la misma definición en varios gestores de colas, `SYSTEM.DEF.CLNTCONN` por ejemplo, el archivo contiene la definición más reciente. Cuando crea un gestor de colas, si se establece **MQCHLLIB** , `SYSTEM.DEF.CLNTCONN` se actualiza en la CCDT. La actualización sobrescribe `SYSTEM.DEF.CLNTCONN` creado por otro gestor de colas. Si ha modificado la definición anterior, las modificaciones se perderán. Por este motivo, debe encontrar alternativas a establecer **MQCHLLIB** como variable de entorno de todo el sistema en el servidor.
7. La opción MQSC y PCF `NOREPLACE` en una definición de conexión con el cliente no comprueba el contenido del archivo CCDT. Una definición de canal de conexión con el cliente con el mismo nombre que se creó anteriormente, pero no por este gestor de colas, se sustituye independientemente de la opción `NOREPLACE`. Si la definición la ha creado anteriormente el mismo gestor de colas, la definición no se sustituye.
8. El mandato, **rcrmqobj** `-t clchltab`, borra y vuelve a crear el archivo CCDT. El archivo se vuelve a crear solo con definiciones de conexión de cliente creadas en el gestor de colas en el que se ejecuta el mandato.
9. Otros mandatos que actualizan la CCDT sólo modifican los canales de conexión de cliente que tienen el mismo nombre de canal. Otros canales de conexión de cliente en el archivo no se modifican.
10. La vía de acceso para **MQCHLLIB** no necesita comillas.

Para obtener más información, consulte Ubicaciones para la CCDT, Acceso de URL a la CCDTy Conexión de aplicaciones cliente a gestores de colas utilizando variables de entorno.

MQCHLTAB

La variable de entorno **MQCHLTAB** especifica el nombre del archivo que contiene la tabla de definición de canal de cliente (CCDT). El nombre de archivo predeterminado es `AMQCLCHL.TAB`.

Para establecer **MQCHLTAB**, utilice uno de estos mandatos:

- Linux AIX En AIX and Linux:

```
export MQCHLTAB=filename
```

- Windows En Windows:

```
SET MQCHLTAB=filename
```

- IBM i En IBM i:

```
ADDENVVAR ENVVAR(MQCHLTAB) VALUE(filename)
```

Por ejemplo:

```
SET MQCHLTAB=ccdf1.tab
```

Del mismo modo que para el cliente, la variable de entorno **MQCHLTAB** en el servidor especifica el nombre de la tabla de definición de canal de cliente.

Para obtener más información, consulte Ubicaciones para la CCDT, Acceso de URL a la CCDTy Conexión de aplicaciones cliente a gestores de colas utilizando variables de entorno.

MQCLNTCF

La variable de entorno **MQCLNTCF** especifica la ubicación del archivo de configuración IBM MQ MQI client . Este archivo contiene información de configuración que utiliza IBM MQ MQI clients.

Puede utilizar la variable de entorno **MQCLNTCF** para modificar la vía de acceso del archivo `mqclient.ini` .

El formato de esta variable de entorno es un URL completo. Esto significa que es posible que el nombre de archivo no sea necesariamente `mqclient.ini`, lo que facilita la colocación del archivo en un sistema de archivos adjunto de red. Para obtener más información, consulte Archivo de configuración del cliente MQI deIBM MQ , mqclient.ini y Ubicación del archivo de configuración del cliente.

MQDOTNET_TRACE_ON

La variable de entorno **MQDOTNET_TRACE_ON** se utiliza para habilitar el rastreo para clientes redistribuibles de IBM MQ .NET . Los valores iguales y menores que 0 no habilitan el rastreo, 1 habilita el rastreo predeterminado y los valores mayores que 1 habilitan el rastreo de detalles.

Para obtener más información, consulte Rastreo de aplicaciones IBM MQ .NET y Rastreo de aplicaciones IBM MQ .NET utilizando variables de entorno.

MQIPADDRV

La variable de entorno **MQIPADDRV** especifica qué protocolo IP utilizar para una conexión de canal. Tiene los valores de serie posibles de `"MQIPADDR_IPV4"` o `"MQIPADDR_IPV6"`. Estos valores tienen el mismo significado que IPv4 y IPv6 en **ALTER QMGR IPADDRV** y el atributo **IPAddressVersion** de la stanza TCP del archivo de configuración de cliente. Si la variable de entorno no está establecida, se presupone `"MQIPADDR_IPV4"` .

Para establecer **MQIPADDRV**, utilice uno de estos mandatos:

- Linux AIX En AIX and Linux:

```
export MQIPADDRV=MQIPADDR_IPV4|MQIPADDR_IPV6"/>
```

- Windows En Windows:

```
SET MQIPADDRV=MQIPADDR_IPV4|MQIPADDR_IPV6
```

- IBM i En IBM i:

```
ADDENVVAR ENVVAR(MQIPADDRV) VALUE(MQIPADDR_IPV4|MQIPADDR_IPV6)
```

MQKEYRPWD

Cuando establece la variable de entorno *MQKEYRPWD* , especifica la contraseña para el repositorio de claves que contiene el certificado digital que pertenece al usuario. Si se utiliza *MQKEYRPWD* , debe cifrar la contraseña antes de establecer el valor de la variable de entorno.

Para establecer *MQKEYRPWD*, utilice uno de estos mandatos:

- Linux AIX En sistemas AIX and Linux:

```
export MQKEYRPWD=passphrase
```

- Windows En sistemas Windows:

```
SET MQKEYRPWD=passphrase
```

- IBM i En IBM i:

```
ADDENVVAR ENVVAR(MQKEYRPWD) VALUE(passphrase)
```

No hay ningún valor predeterminado para esta variable de entorno.

Para obtener más información, consulte

- ALW Suministro de la contraseña del repositorio de claves para un IBM MQ MQI client en AIX, Linux, and Windows y Cifrado de la contraseña del repositorio de claves
- IBM i Suministro de la contraseña del repositorio de claves para un IBM MQ MQI client en IBM i y Cifrado de la contraseña del repositorio de claves.

mqlicense

Linux

En sistemas Linux , puede utilizar la variable de entorno **MQLICENSE** para aceptar o ver una licencia de IBM MQ después de instalar el producto.

Para obtener más información sobre por qué es posible que desee o necesite hacerlo, consulte Aceptación de la licencia en IBM MQ para Linux

La variable de entorno **MQLICENSE** se puede establecer en uno de dos valores:

accept
Acepte la licencia posterior a la instalación.

vista
Visualizar la licencia, si se ha aceptado la licencia.

Para aceptar la licencia posterior a la instalación, utilice este mandato:

```
export MQLICENSE=accept
```

Para ver la licencia, utilice este mandato:

```
export MQLICENSE=view
```

Nota: También puede utilizar los mandatos siguientes para aceptar y visualizar la licencia:

- mqlicense (aceptar licencia posterior a la instalación)
- dspmqlic (visualizar licencia de IBM MQ)

MQMAXERRORLOGSIZE

Multi

La variable de entorno **MQMAXERRORLOGSIZE** especifica el tamaño del registro de errores del gestor de colas que se copia en la copia de seguridad.

Si desea más información, consulte Utilización de registros de errores.

MQNAME

Windows

La variable de entorno **MQNAME** especifica el nombre NetBIOS local que los procesos IBM MQ pueden utilizar. Una conexión NetBIOS se aplica únicamente a un cliente y un servidor que ejecuten Windows.

Para establecer **MQNAME**, utilice este mandato:

```
SET MQNAME=Your_env_Name
```

Por ejemplo:

```
SET MQNAME=CLIENT1
```

Algunas implementaciones de NetBIOS requieren un nombre exclusivo, establecido por **MQNAME**, para cada aplicación si está ejecutando varias aplicaciones IBM MQ simultáneamente en IBM MQ MQI client.

Para obtener más información, consulte Definición del nombre NetBIOS local de IBM MQ.

MQNOREMPOOL

Cuando establece la variable de entorno **MQNOREMPOOL** , desactiva la agrupación de canales y hace que los canales se ejecuten como hebras del escucha.

Para obtener más información, consulte MCATYPE (Tipo de agente de canal de mensajes).

MQPSE_TRACE_LOGFILE

Utilice la variable de entorno **MQPSE_TRACE_LOGFILE** cuando ejecute el programa de ejemplo de salida de publicación AMQSPSE0, que es un programa C de ejemplo de una salida para interceptar una publicación antes de que se entregue a un suscriptor. En el proceso de aplicación que se va a rastrear, esta variable de entorno describe dónde deben grabarse los archivos de rastreo.

Para obtener más información, consulte El programa de ejemplo de salida de publicación.

MQS_AMSCRED_KEYFILE

Puede utilizar la variable de entorno **MQS_AMSCRED_KEYFILE** para alterar temporalmente o proporcionar el archivo de claves inicial para utilizar en tiempo de ejecución de aplicaciones IBM MQ Advanced Message Security (AMS), o al proteger un archivo de configuración de almacén de claves utilizando el mandato **runamscred** .

Para obtener más información, consulte Utilización de almacenes de claves y certificados con AMS y Protección de contraseñas en archivos de configuración de componentes de IBM MQ.

MQS_DISABLE_ALL_INTERCEPT

Puede utilizar la variable de entorno **MQS_DISABLE_ALL_INTERCEPT** para inhabilitar IBM MQ Advanced Message Security (AMS) si se notifica un error 2085 (MQRC_UNKNOWN_OBJECT_NAME) cuando está intentando conectarse a un gestor de colas desde una versión anterior del producto y está utilizando IBM MQ con clientes C nativos.

Nota: Puede utilizar la variable de entorno **MQS_DISABLE_ALL_INTERCEPT** sólo para clientes C. Para los clientes Java , en su lugar debe utilizar la variable de entorno **AMQ_DISABLE_CLIENT_AMS** .

Para obtener más información, consulte Inhabilitación de Advanced Message Security en el cliente.

MQS_IPC_HOST

Puesto que los objetos del sistema de archivos IPC deben distinguirse por el sistema, se añade un subdirectorio para cada sistema en el que se ejecuta el gestor de colas a la vía de acceso del directorio. Si el valor generado del nombre de host crea un problema, puede establecer el nombre de host utilizando la variable de entorno **MQS_IPC_HOST** .

Para obtener más información, consulte Compartir archivos de IBM MQ en Multiplatforms.

MQS_KEYSTORE_CONF

La variable de entorno **MQS_KEYSTORE_CONF** especifica la ubicación del archivo de configuración del almacén de claves para IBM MQ Advanced Message Security (AMS), si el archivo no está en la ubicación predeterminada de *home_directory*/.mqs/keystore.conf.

Para obtener más información, consulte Utilización de almacenes de claves y certificados con AMS.

Si tiene problemas en Managed File Transfer, consulte Resolución de problemas cuando MFT no lee las propiedades del almacén de claves para AMS.

MQS_MQI_XX_ENCODE_CASE_ONE archivo_clave

Cuando establece la variable de entorno **MQS_MQI_KEYFILE** , especifica la ubicación de un archivo de claves inicial que contiene la clave inicial que se utilizará para las operaciones de protección de contraseña. Si no se especifica el archivo de claves inicial, el sistema de protección por contraseña de IBM MQ utiliza la clave inicial predeterminada.

Para establecer **MQS_MQI_KEYFILE**, utilice uno de estos mandatos:

- Linux AIX En sistemas AIX and Linux:

```
export MQS_MQI_KEYFILE=key file location
```

- Windows En sistemas Windows:

```
SET MQS_MQI_KEYFILE=key file location
```

- IBM i En IBM i:

```
ADDENVVAR ENVVAR(MQS_MQI_KEYFILE) VALUE(key file location)
```

Para obtener más información, consulte Suministro de una clave inicial para un IBM MQ MQI client en AIX, Linux, and Windows y Suministro de una clave inicial para un IBM MQ MQI client en IBM i.

MQS_SSLCRYP_KEYFILE

La variable de entorno **MQS_SSLCRYP_KEYFILE** es una forma alternativa de especificar la vía de acceso completa y el nombre del archivo que contiene la clave inicial utilizada para cifrar la contraseña en la serie de configuración de hardware criptográfico PKCS #11 , en lugar de especificarla con el atributo **SSLCryptoHardwareKeyFile** en la stanza SSL de qm.ini. **MQS_SSLCRYP_KEYFILE** tiene una prioridad más alta que el archivo qm.ini , por lo que su valor tiene prioridad sobre cualquier otro valor. Para obtener más información, consulte Clientes de IBM MQ que utilizan hardware criptográfico.

MQS_TRACE_OPTIONS

AIX

Para el rastreo selectivo de componentes en AIX, utilice la variable de entorno **MQS_TRACE_OPTIONS** para activar las funciones de rastreo de parámetros y de detalle alto individualmente.

Nota: Establezca sólo la variable de entorno **MQS_TRACE_OPTIONS** si así se lo ha indicado el servicio de soporte de IBM .

Para obtener más información, consulte Rastreo en AIX and Linux.

MQSERVER

La variable de entorno **MQSERVER** se utiliza para definir un canal mínimo. **MQSERVER** especifica la ubicación del servidor IBM MQ y el método de comunicación que se va a utilizar.

Nota: No puede utilizar **MQSERVER** para definir un canal TLS o un canal con salidas de canal. Para obtener más información sobre cómo definir un canal TLS, consulte Protección de canales con TLS.

Los ejemplos siguientes muestran cómo establecer **MQSERVER**:

- Linux AIX En AIX and Linux:

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

- Windows En Windows:

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

- IBM i En IBM i:

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

Nota:

- El nombre de canal no puede contener el carácter de barra inclinada (/) porque este carácter se utiliza para separar el nombre de canal, el tipo de transporte y el nombre de conexión. Cuando se utiliza la variable de entorno **MQSERVER** para definir un canal de cliente, se utiliza una longitud máxima de mensaje (MAXMSGL) de 100 MB. Por consiguiente, el tamaño máximo de mensaje en vigor para el canal es el valor especificado en el canal SVRCONN en el servidor.
- El tipo de transporte puede ser uno de `LU62` , `TCP` , `NETBIOS`, SPX, en función de la plataforma de cliente de IBM MQ .
- El nombre de conexión debe ser un nombre de red completo., Por ejemplo, `AMACHINE.ACOMPANY.COM(1414)`.
- El nombre de conexión puede ser una lista separada por comas de nombres de conexión. Los nombres de conexiones de la lista se utilizan de un modo similar para varias conexiones en una tabla de conexiones de cliente. La lista de nombres de conexión se puede utilizar como alternativa a los grupos de gestores de colas para especificar varias conexiones para que el cliente las intente. Si está configurando un gestor de colas de varias instancias, puede utilizar una lista de nombres de conexión para especificar distintas instancias de gestor de colas.

Si utiliza la variable de entorno **MQSERVER** para definir el canal entre la máquina IBM MQ MQI client y una máquina servidor, este es el único canal disponible para la aplicación y no se hace referencia a la tabla de definiciones de canal de cliente (CCDT).

Para obtener más información, consulte Creación de un canal de conexión de cliente en el cliente MQI de IBM MQ utilizando MQSERVER.

MQSNOAUT

Aviso: No se recomienda esta funcionalidad.

Cuando establece la variable de entorno **MQSNOAUT** en cualquier valor, inhabilita el gestor de autorizaciones sobre objetos (OAM) e impide cualquier comprobación de seguridad. Esta acción podría resultar adecuada en un entorno de prueba. Esto incluye tanto la autorización como la funcionalidad de autenticación de conexión. TLS, registros de autenticación de canal y salidas de seguridad no se ven afectados.

La variable de entorno **MQSNOAUT** sólo entra en vigor cuando se crea un gestor de colas.

Aviso: Para habilitar el OAM, debe suprimir el gestor de colas, suprimir la variable de entorno y, a continuación, volver a crear el gestor de colas sin especificar **MQSNOAUT**.

Para obtener más información, consulte Impedir comprobaciones de acceso de seguridad en sistemas AIX, Linuxy Windows.

MQSPREFIX

Como alternativa a cambiar el prefijo predeterminado, puede utilizar la variable de entorno **MQSPREFIX** para alterar temporalmente el **DefaultPrefix** para el mandato **crtmqm** .

Para obtener más información, consulte Nombres de archivo deIBM MQ y la stanza AllQueueManagers del archivo mqs.ini.

MQSSLCRYP

ALW

La variable de entorno **MQSSLCRYP** contiene una serie de parámetros que puede utilizar para configurar el hardware de cifrado presente en el sistema.

Los valores permitidos son los mismos que para el campo SSLCryptoHardware de la stanza SSL del archivo de configuración de cliente.

Para establecer **MQSSLCRYP**, utilice uno de estos mandatos:

- Linux AIX En sistemas AIX and Linux:

```
export MQSSLCRYP=string
```

- Windows En sistemas Windows:

```
SET MQSSLCRYP=string
```

Para obtener más información, consulte Configuración para hardware criptográfico en AIX, Linux, and Windows y IBM MQ clients que utilizan hardware criptográfico en Protección de contraseñas en archivos de configuración de componentes de IBM MQ.

MQSSLFIPS

La variable de entorno **MQSSLFIPS** especifica si sólo se van a utilizar algoritmos certificados por FIPS si la criptografía se lleva a cabo en IBM MQ. Puede establecer esta variable de entorno en YES o NO . El valor predeterminado es NO. Estos valores son los mismos que para el parámetro **SSLFIPS** del mandato **ALTER QMGR** .

Para establecer **MQSSLFIPS**, utilice uno de estos mandatos:

- Linux AIX En sistemas AIX and Linux:

```
export MQSSLFIPS=YES|NO
```

- Windows En sistemas Windows:

```
SET MQSSLFIPS=YES|NO
```

- IBM i En IBM i:

```
ADDENVVAR ENVVAR(MQSSLFIPS) VALUE(YES|NO)
```

El uso de algoritmos certificados por FIPS se ve afectado por el uso de hardware criptográfico. Para obtener más información, consulte Especificación de que solo se utilizan CipherSpecs certificadas por FIPS en tiempo de ejecución en el cliente MQI.

MQSSLKEYR

La variable de entorno **MQSSLKEYR** especifica la ubicación del repositorio de claves que contiene el certificado digital que pertenece al usuario.

Especifique la vía de acceso completa y el nombre de archivo del repositorio de claves. Si no se especifica el sufijo de archivo, se presupone que es `.kdb`.

Para establecer **MQSSLKEYR**, utilice uno de estos mandatos:

- Linux AIX En sistemas AIX and Linux:

```
export MQSSLKEYR=pathname
```

- Windows En sistemas Windows:

```
SET MQSSLKEYR=pathname
```

- IBM i En IBM i:

```
ADDENVVAR ENVVAR(MQSSLKEYR) VALUE(pathname)
```

No hay ningún valor predeterminado para esta variable de entorno.

Para obtener más información, consulte el parámetro **SSLKEYR** del mandato **ALTER QMGR** .

MQSSLPROXY

La variable de entorno **MQSSLPROXY** especifica el nombre de host y el número de puerto del servidor proxy HTTP que utilizará GSKit para las comprobaciones de OCSP.

Para establecer **MQSSLPROXY**, utilice uno de estos mandatos:

- Linux AIX En sistemas AIX and Linux:

```
export MQSSLPROXY="string"
```

- Windows En sistemas Windows:

```
SET MQSSLPROXY= string
```

La serie que especifique con **MQSSLPROXY** puede ser el nombre de host o la dirección de red del servidor proxy HTTP que utilizará GSKit para las comprobaciones de OCSP. Esta dirección puede ir seguida de un número de puerto opcional, delimitado mediante paréntesis. Si no especifica el número de puerto, se utiliza el puerto HTTP predeterminado, el 80.

Linux AIX Por ejemplo, en los sistemas AIX and Linux, puede utilizar uno de los mandatos siguientes:

-
```
export MQSSLPROXY="proxy.example.com(80)"
```

-
```
export MQSSLPROXY="127.0.0.1"
```

Para obtener más información, consulte Trabajar con Online Certificate Status Protocol (OCSP).

MQSSLRESET

La variable de entorno **MQSSLRESET** especifica el número de bytes no cifrados enviados y recibidos en un canal TLS antes de que se renegocie la clave secreta TLS. Puede establecerse en un entero comprendido entre 0 y 999.999.999. El valor predeterminado es 0, que indica que las claves secretas no se renegocian nunca. Si especifica una cuenta de restablecimiento de clave secreta TLS entre 1 byte y 32 KB, los canales TLS utilizan una cuenta de restablecimiento de clave secreta de 32 KB. Este número de restablecimiento de clave secreta sirve para evitar restablecimientos de clave excesivos que se producirían para valores de restablecimiento de claves secretas TLS pequeñas.

Para establecer **MQSSLRESET**, utilice uno de estos mandatos:

- Linux AIX En sistemas AIX and Linux:

```
export MQSSLRESET=integer
```

- Windows En sistemas Windows:

```
SET MQSSLRESET=integer
```

- IBM i En IBM i:

```
ADDENVVAR ENVVAR(MQSSLRESET) VALUE(integer)
```

Para obtener más información, consulte Restablecimiento de claves secretas SSL y TLS.

MQSUITEB

ALW

Puede configurar IBM MQ para que funcione en conformidad con el estándar NSA Suite B en las plataformas AIX, Linux, and Windows.

La variable de entorno **MQSUITEB** especifica si se va a utilizar la criptografía compatible con Suite B. Si se va a utilizar la criptografía Suite B, puede especificar la intensidad de la criptografía estableciendo **MQSUITEB** en uno de los valores siguientes:

- NINGUNO
- 128_BIT, 192_BIT
- 128_BIT
- 192_BIT

Puede especificar varios valores utilizando una lista separada por comas. El uso del valor NONE con cualquier otro valor no es válido.

Para obtener más información, consulte Configuración de IBM MQ para Suite B.

MQTCPTIMEOUT

La variable de entorno **MQTCPTIMEOUT** especifica cuánto tiempo espera IBM MQ una llamada de conexión TCP.

ODQ_MSG

Si utiliza un manejador de cola de mensajes no entregados que es diferente de **runmqdlq**, el origen del ejemplo, amqsdlq, está disponible para que lo utilice como base. El ejemplo es como el manejador

de mensajes no entregados proporcionado en el producto, pero el rastreo y el informe de errores son diferentes. Utilice la variable de entorno **ODQ_MSG** para establecer el nombre del archivo que contiene los mensajes de error e información. El archivo que se proporciona se denomina `amqsdlq.msg`.

Para obtener más información, consulte Ejemplo de manejador de cola de mensajes no entregados.

ODQ_TRACE

Si utiliza un manejador de cola de mensajes no entregados que es diferente de **runmqdlq**, el origen del ejemplo, `amqsdlq`, está disponible para que lo utilice como base. El ejemplo es como el manejador de mensajes no entregados proporcionado en el producto, pero el rastreo y el informe de errores son diferentes. Para habilitar el rastreo, establezca la variable de entorno **ODQ_TRACE** en YES o `yes`.

Para obtener más información, consulte Ejemplo de manejador de cola de mensajes no entregados.

WCF_TRACE_ON

Hay dos métodos de rastreo diferentes disponibles para el canal personalizado WCF. Estos dos métodos de rastreo se activan de forma independiente o conjuntamente. Cada método produce su propio archivo de rastreo, de modo que cuando ambos métodos de rastreo están activados, se generan dos archivos de salida de rastreo. Hay cuatro combinaciones para habilitar e inhabilitar los dos métodos de rastreo diferentes. Además de estas combinaciones para habilitar el rastreo de WCF, el rastreo de XMS .NET se puede habilitar utilizando la variable de entorno **WCF_TRACE_ON** .

Para obtener más información, consulte Rastreo del canal personalizado WCF para IBM MQ.

WMQSOAP_HOME

La variable de entorno **WMQSOAP_HOME** se utiliza al completar pasos de configuración adicionales después de que el entorno de alojamiento del servicio .NET SOAP sobre JMS se haya instalado y configurado correctamente en IBM MQ. Es accesible desde un gestor de colas local.

Para obtener más información, consulte WCF client to a .NET service hosted by IBM MQ sample y WCF client to a Axis Java service hosted by IBM MQ sample.

XMS_TRACE_ON, XMS_TRACE_FILE_PATH, XMS_TRACE_FORMAT y XMS_TRACE_SPECIFICATION

Si utiliza IBM MQ classes for XMS .NET Framework, puede configurar el rastreo desde un archivo de configuración de aplicación, así como desde las variables de entorno de XMS . Si utiliza IBM MQ classes for XMS .NET (bibliotecas.NET Standard y .NET 6), debe configurar el rastreo desde las variables de entorno de XMS . Normalmente, el rastreo se utiliza bajo la orientación del equipo de soporte de IBM.

Para habilitar y configurar el rastreo para una aplicación XMS .NET , establezca las siguientes variables de entorno antes de ejecutar la aplicación:

XMS_TRACE_ON
: Si la variable de entorno **XMS_TRACE_ON** está establecida, todo el rastreo está habilitado de forma predeterminada.

XMS_TRACE_FILE_PATH
: La variable de entorno **XMS_TRACE_FILE_PATH** especifica el nombre de vía de acceso completo del directorio en el que se graban los registros de rastreo y FFDC, si desea que estos registros se graben en una ubicación alternativa del directorio de trabajo actual.

XMS_TRACE_FORMAT
: La variable de entorno **XMS_TRACE_FORMAT** especifica el formato de rastreo necesario, que puede ser `BASIC` o `ADVANCED`.

XMS_TRACE_SPECIFICATION
: La variable de entorno **XMS_TRACE_SPECIFICATION** altera temporalmente los valores de rastreo definidos en la sección Rastreo de un archivo de configuración de aplicación. **XMS_TRACE_SPECIFICATION** sólo se aplica a IBM MQ classes for XMS .NET Framework .

Para obtener más información, consulte Rastreo de aplicaciones de XMS .NET y Rastreo de aplicaciones de XMS .NET utilizando variables de entorno de XMS.

Multi Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms

Puede cambiar el comportamiento de IBM MQ o de un gestor de colas individual para que se ajuste a las necesidades de la instalación editando la información en los archivos de configuración (`.ini`). También puede cambiar las opciones de configuración para IBM MQ MQI clients.

Acerca de esta tarea

Puede cambiar la información de configuración de IBM MQ en el nivel del nodo o del gestor de colas cambiando los valores especificados en un conjunto de atributos de configuración (o parámetros) que rigen IBM MQ.

Un archivo de configuración (o archivo de stanza) contiene una o más stanzas, que son grupos de líneas del archivo `.ini` que tienen una función común o definen parte de un sistema, como funciones de registro, funciones de canal y servicios instalables. Puede modificar los atributos de configuración de IBM MQ dentro de los siguientes archivos de configuración:

Archivo de configuración de IBM MQ , `mqs.ini`
El archivo `mqs.ini` afecta a los cambios en el nodo como un todo. Hay un archivo `mqs.ini` para cada instalación de IBM MQ.

Puesto que el archivo de configuración de IBM MQ se utiliza para localizar los datos asociados a gestores de colas, un archivo de configuración inexistente o incorrecto puede hacer que algunos o todos los mandatos MQSC fallen. Además, las aplicaciones no pueden conectarse a un gestor de colas que no esté definido en el archivo de configuración de IBM MQ.

Archivo de configuración de instalación, mqinst.ini
Linux AIX En sistemas AIX and Linux , el archivo de configuración de instalación, `mqinst.ini`, contiene información sobre todas las instalaciones de IBM MQ . El archivo `mqinst.ini` no debe editarse ni referenciarse directamente, puesto que su formato no es fijo y podría cambiar. En su lugar, debe editarlo utilizando mandatos.

Archivo de configuración del gestor de colas, `qm.ini`
El archivo `qm.ini` afecta a los cambios para gestores de colas específicos. Hay un archivo `qm.ini` para cada gestor de colas en el nodo.

Archivo de configuración de IBM MQ MQI client , mqclient.ini
Las opciones de configuración para IBM MQ MQI clients se mantienen por separado, en el archivo de configuración de cliente, que generalmente se denomina `mqclient.ini`.

Archivo de configuración de rastreo de actividad, mqat.ini
El archivo `mqat.ini` se utiliza para configurar el comportamiento de rastreo de actividad.

Puede que necesite editar un archivo de configuración si, por ejemplo:

- Pierde un archivo de configuración. (Recupere la copia de seguridad si puede.)
- Tiene que trasladar uno o más gestores de colas a un nuevo directorio.
- Debe cambiar el gestor de colas predeterminado. Esto podría suceder si suprime accidentalmente el gestor de colas existente.
- El servicio de soporte de IBM le recomienda que lo haga.

Importante: Los cambios que realice en un archivo de configuración normalmente no entrarán en vigor hasta la próxima vez que se inicie el gestor de colas.

Puntos a tener en cuenta sobre la edición de archivos de configuración:

- Los valores de los atributos de un archivo de configuración se establecen de acuerdo con las prioridades siguientes:

  - Los parámetros especificados en la línea de mandatos tienen prioridad sobre los valores definidos en los archivos de configuración.
  - Los valores definidos en los archivos `qm.ini` tienen prioridad sobre los valores definidos en el archivo `mqs.ini` .
- Después de la instalación, puede editar los valores predeterminados en los archivos de configuración de IBM MQ .
- Cuando realice una copia de seguridad de un gestor de colas, recuerde incluir tanto su archivo de configuración (`qm.ini`) como el archivo de configuración central de IBM MQ (`mqs.ini`).
- Si establece un valor incorrecto en un atributo de archivo de configuración, el efecto es el mismo que perder el atributo por completo. El valor se ignora y se emite un mensaje de operador para indicar el problema.
- IBM i En IBM i, los archivos `.ini` son archivos continuos residentes en el IFS.
- Existen varias reglas de sintaxis para el formato del archivo `mqat.ini` . Para obtener más información, consulte Rastreo de actividad de aplicación Configuración del comportamiento de rastreo de actividad utilizando `mqat.ini`.

Procedimiento

1. Antes de editar un archivo de configuración, realice una copia de seguridad para que tenga una copia a la que pueda volver, si es necesario.
2. Edite el archivo de configuración `.ini` de una de las maneras siguientes:
   - Manualmente utilizando un editor de texto estándar. Puede incluir comentarios en los archivos de configuración añadiendo un carácter ";" o "#" antes del texto del comentario. Si desea utilizar un carácter ";" o un carácter "#" sin que represente un comentario, puede añadir un prefijo al carácter con un carácter "\" . A continuación, el carácter se utiliza como parte de los datos de configuración.
   - Automáticamente, utilizando mandatos que cambian la configuración de los gestores de colas en el nodo. Para obtener más información, consulte Referencia de mandatos.

     Windows Por ejemplo, el Windows mandato específico **amqmdain** actualizará automáticamente un subconjunto de las propiedades `qm.ini` . Para obtener más información, consulte amqmdain.
   - Windows Linux En Linux (x86 y x86-64) y Windows, puede actualizar un subconjunto de las propiedades `qm.ini` utilizando IBM MQ Explorer. Para obtener más información, consulte Configuración de IBM MQ utilizando MQ Explorer.

     Nota: Dado que hay implicaciones significativas en el cambio de servicios instalables y sus componentes, los servicios instalables son de sólo lectura en IBM MQ Explorer. Por lo tanto, debe realizar los cambios en los servicios instalables editando el archivo `qm.ini` . Para obtener más información, consulte “Stanza de servicio del archivo qm.ini” en la página 155.

Tareas relacionadas

Administración de IBM MQ

Multi Archivo de configuración de IBM MQ, mqs.ini

El archivo de configuración IBM MQ, `mqs.ini`, contiene información relevante para todos los gestores de colas del nodo. Se crea automáticamente durante la instalación.

Nota: Para obtener más información sobre cómo y cuándo editar el archivo `mqs.ini` y sobre cuándo entran en vigor los cambios que realice en el archivo, consulte “Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms” en la página 95.

Ubicaciones de directorio

Linux AIX En AIX and Linux, el directorio de datos y el directorio de registro son siempre `/var/mqm` y `/var/mqm/log` respectivamente.

Windows En los sistemas Windows, la ubicación del directorio de datos de `mqs.ini` y la ubicación del directorio de registro se almacenan en el registro, ya que su ubicación puede variar. La información de configuración de instalación, que está contenida en `mqinst.ini` en sistemas AIX and Linux, también se encuentra en el registro, ya que no hay ningún archivo `mqinst.ini` en Windows (consulte "Archivo de configuración de instalación, mqinst.ini" en la página 172).

Windows El archivo `mqs.ini` para sistemas Windows lo proporciona la WorkPath especificada en la clave `HKLM\SOFTWARE\IBM\IBM MQ`. Contiene:

- Los nombres de los gestores de colas
- El nombre del gestor de colas predeterminado
- La ubicación de los archivos asociados a cada uno de ellos.

IBM i En IBM i, el archivo `mqs.ini` se almacena en `/QIBM/UserData/mqm`. El archivo contiene:

- Los nombres de los gestores de colas.
- El nombre del gestor de colas predeterminado.
- La ubicación de los archivos asociados con cada gestor de colas.
- La información que identifica todas las salidas de API (consulte Configuración de las salidas de API si desea más información).

En particular, el archivo `mqs.ini` se utiliza para localizar los datos asociados con cada gestor de colas.

Archivo `mqs.ini` de ejemplo para AIX and Linux

Linux AIX

```
#*******************************************************************#
#* Module Name: mqs.ini                                            *#
#* Type       : IBM MQ Machine-wide Configuration File             *#
#* Function   : Define IBM MQ resources for an entire machine      *#
#*******************************************************************#
#* Notes      :                                                    *#
#* 1) This is the installation time default configuration          *#
#*                                                                 *#
#*******************************************************************#
AllQueueManagers:
#*******************************************************************#
#* The path to the qmgrs directory, below which queue manager data *#
#* is stored                                                       *#
#*******************************************************************#
DefaultPrefix=/var/mqm

LogDefaults:
   LogPrimaryFiles=3
   LogSecondaryFiles=2
   LogFilePages=4096
   LogType=CIRCULAR
   LogBufferPages=0
   LogDefaultPath=/var/mqm/log

QueueManager:
   Name=saturn.queue.manager
   Prefix=/var/mqm
   Directory=saturn!queue!manager
   InstallationName=Installation1

QueueManager:
   Name=pluto.queue.manager
   Prefix=/var/mqm
   Directory=pluto!queue!manager
   InstallationName=Installation2
```

```
DefaultQueueManager:
   Name=saturn.queue.manager

ApiExitTemplate:
   Name=OurPayrollQueueAuditor
   Sequence=2
   Function=EntryPoint
   Module=/usr/ABC/auditor
   Data=123

ApiExitCommon:
   Name=MQPoliceman
   Sequence=1
   Function=EntryPoint
   Module=/usr/MQPolice/tmqp
   Data=CheckEverything
```

Archivo `mqs.ini` de ejemplo para Windows

Windows

```
#********************************************************************#
#* Module Name: mqs.ini                                             *#
#* Type       : IBM MQ Machine-wide Configuration File              *#
#* Function   : Define IBM MQ resources for an entire machine       *#
#********************************************************************#
#* Notes      :                                                     *#
#* 1) This is the installation time default configuration           *#
#*                                                                  *#
#********************************************************************#
AllQueueManagers:
#********************************************************************#
#* The path to the qmgrs directory, below which queue manager data  *#
#* is stored                                                        *#
#********************************************************************#
DefaultPrefix=C:\ProgramData\IBM\MQ

LogDefaults:
   LogPrimaryFiles=3
   LogSecondaryFiles=2
   LogFilePages=4096
   LogType=CIRCULAR
   LogBufferPages=0
   LogDefaultPath=C:\ProgramData\IBM\MQ\log

QueueManager:
   Name=saturn.queue.manager
   Prefix=C:\ProgramData\IBM\MQ
   Directory=saturn!queue!manager
   InstallationName=Installation1

QueueManager:
   Name=pluto.queue.manager
   Prefix=C:\ProgramData\IBM\MQ
   Directory=pluto!queue!manager
   InstallationName=Installation2

DefaultQueueManager:
   Name=saturn.queue.manager

ApiExitTemplate:
   Name=OurPayrollQueueAuditor
   Sequence=2
   Function=EntryPoint
   Module=C:\usr\ABC\auditor
   Data=123

ApiExitCommon:
   Name=MQPoliceman
   Sequence=1
   Function=EntryPoint
   Module=C:\usr\MQPolice\tmpq
   Data=CheckEverything
```

Archivo `mqs.ini` de ejemplo para IBM i

IBM i

```
#********************************************************************#
#* Module Name: mqs.ini                                             *#
#* Type       : IBM MQ Configuration File                           *#
#* Function   : Define IBM MQ resources for the node                *#
#*                                                                  *#
#********************************************************************#
#* Notes      :                                                     *#
#* 1) This is an example IBM MQ configuration file                  *#
#*                                                                  *#
#********************************************************************#
AllQueueManagers:
#********************************************************************#
#* The path to the qmgrs directory, within which queue manager data *#
#* is stored                                                        *#
#********************************************************************#
DefaultPrefix=/QIBM/UserData/mqm

QueueManager:
Name=saturn.queue.manager
Prefix=/QIBM/UserData/mqm
Library=QMSATURN.Q
Directory=saturn!queue!manager

QueueManager:
Name=pluto.queue.manager
Prefix=/QIBM/UserData/mqm
Library=QMPLUTO.QU
Directory=pluto!queue!manager

DefaultQueueManager:
Name=saturn.queue.manager
```

Notas:

1. IBM MQ en el nodo utiliza las ubicaciones predeterminadas para los gestores de colas y los diarios.
2. El gestor de colas saturn.queue.manager es el gestor de colas predeterminado del nodo. El directorio de los archivos asociados a este gestor de colas se ha transformado automáticamente en un nombre de archivo válido para el sistema de archivos.
3. Puesto que el archivo de configuración de IBM MQ se utiliza para localizar los datos asociados a gestores de colas, un archivo de configuración inexistente o incorrecto puede hacer que algunos o todos los mandatos de IBM MQ fallen. Además, las aplicaciones no pueden conectarse a un gestor de colas que no esté definido en el archivo de configuración de IBM MQ.

`mqs.ini` stanzas

Atención: Este tema enlaza con más información sobre las stanzas del archivo `mqs.ini` . Cada stanza contiene información sobre los parámetros de dicha stanza.

Multi Resumen de stanzas y atributos del archivo mqs.ini

Un resumen de los atributos de las stanzas del archivo de configuración de IBM MQ, `mqs.ini`, con enlaces a más información.

Tabla 10. Stanzas del archivo mqs.ini

Stanza y atributos	Descripción de atributos
Stanza AllQueueManagers	
DefaultPrefix	La vía de acceso al directorio `qmgrs`, en el que se guardan los datos del gestor de colas.
ALW DefaultEphemeralPrefijo	La vía de acceso del directorio, dentro del cual se conservan los datos efímeros del gestor de colas.

Tabla 10. Stanzas del archivo mqs.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
Multi ConvEBCDICNewline	Cómo IBM MQ va a convertir el carácter NL EBCDIC en formato ASCII
Stanza ApiExitCommon y stanza ApiExitTemplate	
Nombre	El nombre que describe la salida de API que se ha pasado en el campo ExitInfoName de la estructura MQAXP.
Función	El nombre del punto de entrada de la función al módulo que contiene el código de la salida de API.
Módulo	El módulo que contiene el código de la salida de API.
Datos	Los datos que se han de pasar a la salida de API en el campo ExitData de la estructura MQAXP.
Sequence	La secuencia en que se llama a esta salida de API es relativa para las otras salidas de API.
stanza DefaultQueueManager	
Nombre	El nombre del gestor de colas que procesa los mandatos para los cuales no se ha especificado explícitamente un nombre de gestor de colas.
sección ExitProperties	
CLWLMode	Indica si la salida de carga de trabajo del clúster (CLWL) se ejecuta en la modalidad FAST o en la modalidad SAFE.
sección LogDefaults	
LogPrimaryFiles	Los archivos de anotaciones asignados cuando se crea el gestor de colas.
LogSecondaryFiles	Los archivos de anotaciones que se asignan cuando se agotan los archivos primarios.
LogFilePages	El número de páginas de archivo de registro. (El tamaño del archivo de registro se especifica en unidad de páginas de 4 KB.)
LogType	El tipo de registro que va a utilizar el gestor de colas (circular o lineal).
LogBufferPages	La cantidad de memoria asignada a los registros de almacenamiento intermedio para grabación, especificando el tamaño de los almacenamientos intermedios en unidades de páginas de 4 KB.
LogDefaultPath	El directorio en el que residen los archivos de registro de un gestor de colas.
LogWriteIntegrity	El método que utiliza el registrador de anotaciones para grabar los registros de anotaciones de forma fiable.
sección QueueManager	
Nombre	Nombre del gestor de colas.
Prefix	Indica dónde están almacenados los archivos del gestor de colas.

Tabla 10. Stanzas del archivo mqs.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
Directorio	El nombre del subdirectorio bajo el directorio `prefix\QMGRS` donde se almacenan los archivos del gestor de colas.
DataPath	Se ha creado una vía de acceso de datos explícita proporcionada con el gestor de colas; esto sobrescribe Prefijo y Directorio como vía de acceso a los datos del gestor de colas.
InstallationName	Nombre de la instalación de IBM MQ asociada a este gestor de colas.
EphemeralPrefix	Donde se almacenan los datos efímeros del gestor de colas.

Multi Stanza AllQueueManagers del archivo mqs.ini

La stanza AllQueueManagers puede especificar la vía de acceso al directorio `qmgrs` donde se almacenan los archivos asociados con un gestor de colas, la vía de acceso a la biblioteca ejecutable y el método para convertir datos con formato EBCDIC a formato ASCII.

Utilice la stanza AllQueueManagers del archivo `mqs.ini` para especificar la información sobre todos los gestores de colas.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades IBM MQ Explorer `General` y `Extended` IBM MQ .

DefaultPrefix=*nombre_directorio*
Este atributo especifica la vía de acceso al directorio `qmgrs`, en el que se guardan los datos del gestor de colas.

Si cambia el prefijo predeterminado del gestor de colas, reproduzca la estructura de directorios creada en el momento de la instalación. Concretamente, debe crear la estructura qmgrs. Detenga IBM MQ antes de cambiar el prefijo predeterminado y reinicie IBM MQ sólo después de haber trasladado las estructuras a la nueva ubicación y de haber cambiado el prefijo predeterminado.

Nota: ALW No suprima el directorio `/var/mqm/errors` en sistemas AIX and Linux, ni el directorio `\errors` en sistemas Windows.

Como alternativa a cambiar el prefijo predeterminado, puede utilizar la variable de entorno **MQSPREFIX** para alterar temporalmente el **DefaultPrefix** para el mandato **crtmqm** .

Debido a las restricciones del sistema operativo, mantenga la vía de acceso suministrada lo suficientemente corta para que la suma de la longitud de la vía de acceso y cualquier nombre del gestor de colas tenga una longitud máxima de 70 caracteres.

Multi **DefaultEphemeralPrefix= *nombre_directorio***
Este atributo especifica la vía de acceso del directorio, dentro del cual se conservan los datos efímeros del gestor de colas como, por ejemplo, sockets de IPC, y solo se utiliza para establecer **EphemeralPrefix** de un gesto de colas cuando se crea un gestor de colas. Además, debe crear el directorio usted mismo si cambia el valor predeterminado. Debe crear el directorio de datos efímeros con permisos que permitan un acceso de escritura al grupo IBM MQ a dicho directorio.

Como alternativa a cambiar el archivo `mqs.ini` , puede utilizar la variable de entorno **MQ_EPHEMERAL_PREFIX** para alterar temporalmente **DefaultEphemeralPrefix** para el mandato **crtmqm** .

Debido a las restricciones del sistema operativo, el prefijo efímero predeterminado está restringido a:

- Linux AIX 12 caracteres en plataformas AIX and Linux.
- IBM i 24 caracteres en IBM i.

MQ Appliance **DefaultEphemeralPrefix** no está soportado en IBM MQ Appliance.

Multi **ConvEBCDICNewline=NL_TO_LF|TABLE|ISO**
Las páginas de códigos EBCDIC contienen un carácter de línea nueva (NL) para el que las páginas de códigos ASCII no tienen soporte, aunque algunas variantes ISO de ASCII contienen un equivalente. Utilice el atributo **ConvEBCDICNewline** para especificar cómo se va a convertir IBM MQ al carácter EBCDIC NL en formato ASCII.

IBM i En IBM MQ for IBM i, se considera que CCSID 1253 es un CCSID ISO, y NL_TO_LF afecta a ambas conversiones, ISO y ASCII.

z/OS El atributo **ConvEBCDICNewline** no está disponible en z/OS. El comportamiento en z/OS es equivalente a `ConvEBCDICNewline=TABLE`. Tenga en cuenta que el valor predeterminado en otras plataformas podría ser diferente.

NL_TO_LF
Convertir el carácter NL de EBCDIC (X'15') en el carácter de salto de línea LF de ASCII, (X'0A'), para todas las conversiones de EBCDIC a ASCII.

`NL_TO_LF` es el valor predeterminado.

TABLE
Convertir el carácter NL de EBCDIC de acuerdo con las tablas de conversión utilizadas en su plataforma para todas las conversiones de EBCDIC a ASCII.

El efecto de este tipo de conversión puede variar de una plataforma a otra y de un idioma a otro; incluso en una misma plataforma, el comportamiento puede variar si se utilizan distintos CCSID.

ISO
Convertir:

- Identificadores de juego de caracteres codificados ISO utilizando el método TABLE
- Todos los demás identificadores de juego de caracteres utilizando el método `NL_TO_CF`.

En la Tabla 11 en la página 102 se muestran los CCSID de ISO posibles.

Tabla 11. Lista de CCSID de ISO posibles

CCSID	Página de códigos
819	ISO8859-1
912	ISO8859-2
915	ISO8859-5
1089	ISO8859-6
813	ISO8859-7
916	ISO8859-8
920	ISO8859-9
1051	roman8

Si el identificador de juego de caracteres codificados ASCII no es un subconjunto ISO, **ConvEBCDICNewline** toma como valor predeterminado `NL_TO_LF`.

Desde IBM MQ 9.1.0 Fix Pack 2 y IBM MQ 9.1.2, puede utilizar la variable de entorno **AMQ_CONVEBCDICNEWLINE** en lugar del atributo de la stanza **ConvEBCDICNewline**, por ejemplo,

para proporcionar la funcionalidad **ConvEBCDICNewline** en el lado del cliente en situaciones en las que no se puede utilizar el archivo `mqs.ini`. La variable de entorno toma los mismos valores (NL_TO_LF, TABLE o ISO) que el atributo **ConvEBCDICNewline**. El atributo de stanza tiene prioridad si se establecen tanto el atributo como la variable de entorno.

Multi Stanzas ApiExitCommon y ApiExitTemplate del archivo mqs.ini

La plantilla ApiExity las stanzas ApiExitCommon identifican rutinas de salida de API para todos los gestores de colas.

Utilice las stanzas ApiExitTemplate y ApiExitCommon en el archivo `mqs.ini` para identificar rutinas de salida de API para todos los gestores de colas. (Para identificar rutinas de salida de API para gestores de colas individuales, utilice la stanza local ApiExit, tal como se describe en "Stanza ApiExitLocal del archivo qm.ini" en la página 122.)

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades de IBM MQ Explorer `Exits` IBM MQ .

Windows En Windows, también puede utilizar el mandato **amqmdain** para cambiar las entradas de las salidas de API.

Para obtener más información sobre cómo utilizar estos atributos, consulte Configuración de salidas de API.

Name=nombre_ApiExit
El nombre que describe la salida de API que se ha pasado en el campo ExitInfoName de la estructura MQAXP.

Este nombre debe ser exclusivo, con un máximo de 48 caracteres de longitud y sólo puede contener caracteres válidos para los nombres de objetos IBM MQ (por ejemplo, nombres de colas).

Function=nombre_función
El nombre del punto de entrada de la función al módulo que contiene el código de la salida de API. Este punto de entrada es la función MQ_INIT_EXIT.

La longitud de este campo está limitada a MQ_EXIT_NAME_LENGTH.

Module=nombre_módulo
El módulo que contiene el código de la salida de API.

Si este campo contiene el nombre de vía de acceso completo del módulo, se utilizará tal y como está. Si este campo contiene solo el nombre de módulo, el módulo se encuentra utilizando el atributo **ExitsDefaultPath** en la stanza `ExitPath` del archivo `qm.ini`.

En plataformas que dan soporte a bibliotecas con hebras independientes, debe proporcionar tanto una versión sin hebras como una versión con hebras del módulo de salida de API. La versión con hebras debe tener un sufijo `_r`. La versión con hebras del apéndice de aplicación de IBM MQ añade implícitamente un sufijo `_r` al nombre de módulo especificado antes de cargarlo.

La longitud de este campo está limitada a la longitud máxima de vía de acceso a la que dé soporte la plataforma.

Data=nombre_datos
Los datos que se han de pasar a la salida de API en el campo ExitData de la estructura MQAXP.

Si incluye este atributo, se suprimirán los espacios en blanco iniciales y de cola, la serie restante se truncará a 32 caracteres y el resultado se pasará a la salida. Si omite este atributo, se pasará el valor predeterminado de 32 espacios en blanco.

La longitud máxima de este campo es de 32 caracteres.

Sequence=número_secuencia
La secuencia en que se llama a esta salida de API es relativa para las otras salidas de API. Se llama antes a una salida con un número de secuencia bajo que a una salida con un número de secuencia más alto. No es necesario que los números de secuencia de las salidas sean contiguos. Una secuencia de 1, 2, 3 tiene el mismo resultado que una secuencia de 7, 42, 1096. Si dos salidas tienen el mismo número de secuencia, el gestor de colas decide a cuál de ellos llamará en primer lugar. Puede saber

a cuál se ha llamado después del suceso, colocando la hora o un marcador en ExitChainArea, que se indica mediante ExitChainAreaPtr en MQAXP, o escribiendo su propio archivo de anotaciones.

Este atributo es un valor numérico sin signo.

Multi Stanza DefaultQueueManager del archivo mqs.ini

La stanza DefaultQueuedel gestor especifica el gestor de colas predeterminado para el nodo.

Utilice la stanza DefaultQueuedel archivo `mqs.ini` para especificar el gestor de colas predeterminado.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice IBM MQ Explorer la página de propiedades de `General` IBM MQ .

Name=*gestor_colas_predeterminado*
El gestor de colas predeterminado procesa todos los mandatos para los que no se especifica explícitamente un nombre de gestor de colas. El atributo **`DefaultQueueManager`** se actualiza automáticamente cuando se crea un nuevo gestor de colas predeterminado. Si crea accidentalmente un nuevo gestor de colas predeterminado y luego desea revertir al original, deberá modificar manualmente el atributo **`DefaultQueueManager`**.

Multi Stanza ExitProperties del archivo mqs.ini

La stanza ExitProperties especifica las opciones de configuración utilizadas por los programas de salida del gestor de colas.

Utilice la stanza ExitProperties del archivo `mqs.ini` para especificar las opciones de configuración utilizadas por los programas de salida del gestor de colas.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades de IBM MQ Explorer `Extended` IBM MQ .

CLWLMode = SAFE (valor predeterminado) | FAST
La salida de carga de trabajo de clúster (CLWL) le permite especificar qué cola de clúster del clúster se debe abrir en respuesta a una llamada MQI (por ejemplo, `MQOPEN`, `MQPUT`). La salida CLWL se ejecuta en modalidad FAST o en modalidad SAFE en función del valor que especifique en el atributo **`CLWLMode`** . Si omite el atributo **`CLWLMode`** , la salida de carga de trabajo del clúster se ejecuta en modalidad SAFE.

SAFE
Ejecutar la salida CLWL en un proceso distinto al del gestor de colas. Este es el valor predeterminado.

Si surge algún problema con la salida CLWL escrita por el usuario mientras se está ejecutando en modalidad SAFE, se producirá lo siguiente:

- El proceso del servidor CLWL (amqzlwa0) no se ejecutará correctamente.
- El gestor de colas reiniciará el proceso del servidor CLWL.
- El error se indicará en los registros de error. Si hay una llamada MQI en proceso, se recibirá una notificación en forma de código de retorno.

Se mantiene la integridad del gestor de colas.

Nota: La ejecución de la salida CLWL en un proceso independiente puede afectar al rendimiento.

FAST
Ejecutar la salida de clúster incorporada en el proceso del gestor de colas.

Especificar esta opción mejora el rendimiento al evitar los costes de conmutación de proceso que implica la ejecución en modalidad SAFE, pero esto se produce a expensas de la integridad del gestor de colas. Sólo debe ejecutar la salida CLWL en modalidad FAST si está convencido de que no hay problemas con la salida CLWL y está especialmente preocupado por el rendimiento.

Si surge algún problema cuando la salida CLWL está ejecutándose en modalidad FAST, el gestor de colas no se ejecutará correctamente y correrá el riesgo de comprometer la integridad del gestor de colas.

Multi Stanza LogDefaults del archivo mqs.ini

La stanza LogDefaults especifica información sobre los valores predeterminados de registro para todos los gestores de colas.

Utilice la stanza LogDefaults del archivo `mqs.ini` para especificar información sobre los valores predeterminados de registro para todos los gestores de colas.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades de IBM MQ Explorer `Default log settings` IBM MQ .

Si necesita un valor no predeterminado, debe especificar explícitamente ese valor en la stanza LogDefaults .

Si la stanza LogDefaults no existe, se utilizan los valores predeterminados de IBM MQ . Los atributos de anotaciones se utilizan como valores predeterminados al crear un gestor de colas, pero pueden alterarse temporalmente si se especifican los atributos de anotaciones en el mandato **crtmqm**. Para obtener más información sobre este mandato, consulte **crtmqm**.

Después de haber creado un gestor de colas, los atributos de registro de dicho gestor de colas se obtienen de los valores que se describen en "Stanza de registro del archivo qm.ini" en la página 145.

Nota: La stanza LogDefaults proporcionada para una nueva instalación de IBM MQ no contiene ningún valor explícito para los atributos. La falta de un atributo significa que el valor predeterminado de este valor se utiliza después de la creación de un nuevo gestor de colas. Los valores predeterminados para la stanza LogDefaults se muestran en "Archivo mqs.ini de ejemplo para AIX and Linux" en la página 97 y "Archivo mqs.ini de ejemplo para Windows" en la página 98. Un valor de cero para el atributo `LogBufferPages` significa 512.

El prefijo predeterminado, que se especifica en "Stanza AllQueueManagers del archivo mqs.ini" en la página 101, y la vía de acceso de registro especificada para el gestor de colas concreto, que se especifica en "Stanza de registro del archivo qm.ini" en la página 145, permiten que el gestor de colas y su registro estén en unidades físicas diferentes. Éste es el método aconsejado aunque, de forma predeterminada, se encuentren en la misma unidad.

Para obtener información sobre el cálculo de tamaños de anotaciones, consulte el apartado "Cálculo del tamaño del registro" en la página 680.

Nota: Los límites indicados en la siguiente lista de parámetros son límites establecidos por IBM MQ. Los límites del sistema operativo podrían reducir el tamaño de registro máximo posible.

LogPrimaryFiles = 3 (valor predeterminado) |*2-254* (Windows) |*2-510* (AIX and Linux)

Los archivos de anotaciones asignados cuando se crea el gestor de colas.

El número mínimo de archivos de registro primarios que puede tener es 2 y el máximo es 254 en Windows, o 510 en AIX and Linux. El valor predeterminado es 3.

El número total de archivos de registro primarios y secundarios no debe superar los 255 en Windows o los 511 en AIX and Linux, y no debe ser inferior a 3.

Cuando se crea o inicia el gestor de colas, se examina el valor. Puede cambiarlo después de haber creado el gestor de colas. No obstante, si modifica el valor, el cambio no entra en vigor hasta que se reinicia el gestor de colas, y es posible que el efecto no sea inmediato.

LogSecondaryFiles = 2 (valor predeterminado) |*1-253* (Windows) |*1-509* (AIX and Linux)

Los archivos de anotaciones que se asignan cuando se agotan los archivos primarios.

El número mínimo de archivos de registro secundarios es 1 y el máximo es 253 en Windows, o 509 en AIX and Linux. El valor predeterminado es 2.

El número total de archivos de registro primarios y secundarios no debe superar los 255 en Windows o los 511 en AIX and Linux, y no debe ser inferior a 3.

El valor se examina cuando se inicia el gestor de colas. Puede modificar este valor, pero los cambios no surtirán efecto hasta que se reinicie el gestor de colas, y es posible que el efecto no sea inmediato.

LogFilePages=*número*
Los datos de las anotaciones se guardan en una serie de archivos llamados archivos de anotaciones. El tamaño del archivo de registro se especifica en unidades de páginas de 4 KB.

El número predeterminado de páginas de archivo de registro es 4096, lo que da un tamaño de archivo de registro de 16 MB.

En AIX and Linux, el número mínimo de páginas de archivo de registro es 64, y en Windows, el número mínimo de páginas de archivo de registro es 32; en ambos casos el número máximo es 65 535.

Nota: El tamaño de los archivos de registro especificado durante la creación del gestor de colas no se puede modificar para un gestor de colas.

LogType = CIRCULAR (valor predeterminado) | LINEAL
El tipo de anotaciones que se utilizará. El valor predeterminado es CIRCULAR.

CIRCULAR
Inicie la recuperación de reinicio utilizando el registro para retrotraer las transacciones que estaban en curso cuando se detuvo el sistema.

Consulte "Tipos de registro" en la página 674 para ver una explicación completa del registro circular.

LINEAR
Este valor permite efectuar tanto la recuperación de reinicio como la recuperación desde soporte o por repetición de actualizaciones (creando los datos perdidos o dañados mediante la reproducción del contenido del registro).

En el apartado "Tipos de registro" en la página 674 puede ver una explicación completa sobre las anotaciones cronológicas lineales.

Si desea cambiar el valor predeterminado, puede editar el atributo LogType o especificar las anotaciones cronológicas lineales mediante el mandato **crtmqm**.

Puede cambiar el método de registro después de que se haya creado un gestor de colas. Para obtener más información, consulte migmqlog.

LogBufferPages=0 (valor predeterminado) |*0-4096*
La cantidad de memoria asignada a los registros de almacenamiento intermedio para grabación, especificando el tamaño de los almacenamientos intermedios en unidades de páginas de 4 KB.

El número mínimo de páginas de almacenamiento intermedio es de 18 y el número máximo es de 4.096. Los almacenamientos intermedios más grandes dan como resultado un rendimiento superior, especialmente para mensajes grandes.

Si especifica 0 (el valor predeterminado), el gestor de colas selecciona el tamaño que es 512 (2048 KB).

Si especifica un número entre 1 y 17, el gestor de colas toma de forma predeterminada el valor de 18 (72 KB). Si especifica un número dentro del rango de 18 a 4096, el gestor de colas utiliza el número especificado para definir la memoria asignada.

LogDefaultPath=*nombre_directorio*
El directorio en el que residen los archivos de registro de un gestor de colas. El directorio se encuentra en un dispositivo local en el que el gestor de colas pueda grabar y, preferiblemente, debe estar una unidad que no sea la que contiene las colas de mensajes. Especificando una unidad distinta se consigue una protección adicional por si se produce una anomalía en el sistema.

El valor predeterminado es:

- Windows *DefaultPrefix*\log para IBM MQ for Windows donde *DefaultPrefix* es el valor especificado en el atributo DefaultPrefix en la página de propiedades de All Queue Managers IBM MQ . Este valor se establece durante la instalación.
- Linux AIX /var/mqm/log para sistemas AIX and Linux.

De forma alternativa, puede especificar el nombre de un directorio en el mandato **crtmqm** utilizando el distintivo **-ld** . Al crear un gestor de colas, también se crea un directorio debajo del directorio del gestor de colas, que se utiliza para contener los archivos de registros. El nombre de este directorio se basa en el nombre del gestor de colas. Esto asegura que la vía de acceso del archivo de registro sea exclusiva y que cumpla con los límites establecidos para la longitud de nombres de directorios.

Si no especifica **-ld** en el mandato **crtmqm** , se utiliza el valor del atributo **LogDefaultPath** en el archivo mqs.ini .

El nombre del gestor de colas se añade al nombre del directorio para asegurar que varios gestores de colas utilicen directorios de registros diferentes.

Cuando se ha creado el gestor de colas, se crea un valor **LogPath** en los atributos de registro de la información de configuración que indica el nombre completo del directorio que corresponde al registro del gestor de colas. Este valor se utiliza para localizar el registro cuando se inicia o se suprime el gestor de colas.

LogWriteIntegrity =SingleWrite|DoubleWrite|TripleWrite (valor predeterminado)
El método que utiliza el registrador de anotaciones para grabar los registros de anotaciones de forma fiable.

TripleWrite (valor predeterminado)

Observe que puede seleccionar DoubleWrite, pero si lo hace, el sistema los interpreta como TripleWrite.

SingleWrite
Debe utilizar SingleWrite, solo si el sistema de archivos y el dispositivo que aloja el registro de recuperación de IBM MQ garantiza de forma explícita la atomicidad de escrituras de 4 KB.

Es decir, cuando una escritura de una página de 4KB falla por algún motivo, los dos únicos estados posibles son la imagen anterior o la imagen posterior. No debería ser posible ningún estado intermedio.

Nota: Si hay suficiente simultaneidad en la carga de trabajo persistentes, hay un mínimo beneficio potencial en establecer cualquier otra cosa que el valor predeterminado, TripleWrite.

Para obtener más información, consulte “LogWriteIntegridad-utilizando SingleWrite o TripleWrite” en la página 148.

Multi Stanza QueueManager del archivo mqs.ini

La stanza QueueManager especifica la ubicación del directorio del gestor de colas.

Hay una stanza QueueManager para cada gestor de colas. Los atributos de esta stanza especifican el nombre del gestor de colas y el nombre del directorio que contiene los archivos asociados con ese gestor de colas. El nombre del directorio se basa en el nombre del gestor de colas, pero se transforma si el nombre del gestor de colas no es un nombre de archivo válido. Para obtener más información sobre la transformación de nombres, consulte Descripción de los nombres de archivo de IBM MQ.

Name=*nombreGC*
Nombre del gestor de colas.

Prefix=*prefijo*
Indica dónde están almacenados los archivos del gestor de colas. De forma predeterminada, este valor es el mismo que el valor especificado en el atributo **DefaultPrefix** de la stanza Todos los gestores de colas del archivo mqs.ini .

Directory=*nombre*
El nombre del subdirectorio bajo el directorio *prefix*\QMGRS donde se almacenan los archivos del gestor de colas. Este nombre se basa en el nombre del gestor de colas, pero puede transformarse si hay algún nombre duplicado o si el nombre del gestor de colas no es un nombre de archivo válido.

DataPath=*vía_acceso*
Una vía de acceso de datos explícita proporcionada cuando se creó el gestor de colas, altera temporalmente **Prefix** y **Directory** como la vía de acceso a los datos del gestor de colas.

InstallationName=*nombre*
Nombre de la instalación de IBM MQ asociada a este gestor de colas. Los mandatos de esta instalación deben utilizarse al interactuar con este gestor de colas.

IBM i **Library=*nombre***
El nombre de la biblioteca en la que se almacenan los objetos de IBM i correspondientes a este gestor de colas, como son por ejemplo, los diarios y los receptores de diario. Este nombre se basa en el nombre del gestor de colas, pero puede transformarse si el nombre está repetido o si el nombre del gestor de colas no es un nombre de biblioteca válido.

EphemeralPrefix= *nombre*
Donde se almacenan los datos efímeros del gestor de colas.

De forma predeterminada, este valor no está presente, lo que significa que los datos se almacenan debajo de la ubicación del prefijo.

El valor se establece a partir del valor de la variable de entorno **MQ_EPHEMERAL_PREFIX** , o del atributo **DefaultEphemeralPrefix** de la stanza AllQueueManagers en el archivo `mqs.ini` , cuando se crea el gestor de colas.

IBM i Debido a las restricciones del sistema operativo, el prefijo efímero predeterminado está restringido a 24 caracteres en IBM i.

Tareas relacionadas

“Asociación de un gestor de colas con una instalación” en la página 487
Cuando se crea un gestor de colas, éste se asocia automáticamente a la instalación que ha emitido el mandato **crtmqm**. En AIX, Linux, and Windows, puede cambiar la instalación asociada a un gestor de colas mediante el mandato **setmqm**.

Windows Interfaz avanzada de configuración y energía (ACPI)

Windows da soporte al estándar Interfaz avanzada de configuración y energía (ACPI). Esto permite que los usuarios de Windows que tienen habilitado el hardware de la ACPI detengan y reinicien canales cuando el sistema entra en modalidad de suspensión o se recupera de la misma.

Utilice la página de propiedades ACPI IBM MQ de IBM MQ Explorer para especificar cómo se comportará IBM MQ cuando el sistema reciba una solicitud de suspensión.

Tenga en cuenta que los valores especificados en la página de propiedades ACPI de IBM MQ sólo se aplican cuando el Supervisor de alertas está en ejecución. El icono del Supervisor de alertas se muestra en la barra de tareas cuando está en ejecución.

DoDialog=Y | N
Muestra el diálogo en el momento en que se produce una petición de suspensión.

DenySuspend=Y | N
Rechaza la petición de suspensión. Se utiliza si DoDialog=N, o si DoDialog=Y y no se puede mostrar un diálogo, por ejemplo porque la tapa del ordenador portátil está cerrada.

CheckChannelsRunning=Y | N
Comprueba si hay algún canal ejecutándose. El resultado puede determinar el resultado de otros valores.

La tabla siguiente describe el efecto de cada combinación de estos parámetros:

DoDialog	DenySuspend	CheckChannels Running	Acción

N	N	N	Aceptar la solicitud de suspensión.
N	N	Y	Aceptar la solicitud de suspensión.
N	Y	N	Denegar la petición de suspensión.
N	Y	Y	Si hay algún canal en ejecución, denegar la solicitud de suspensión; de lo contrario, aceptar la solicitud.
Y	N	N	Mostrar el diálogo (vea la Nota; aceptar la solicitud de suspensión). Este es el valor predeterminado.
Y	N	Y	Si no hay ningún canal en ejecución, aceptar la solicitud de suspensión; si hay canales en ejecución, mostrar el diálogo (vea la Nota; aceptar la solicitud).
Y	Y	N	Mostrar el diálogo (Nota; denegar la solicitud de suspensión).
Y	Y	Y	Si no hay ningún canal en ejecución, aceptar la solicitud de suspensión; si hay canales en ejecución, mostrar el diálogo (Nota; denegar la solicitud).

Nota: En los casos en que la acción sea mostrar el diálogo, si no se puede mostrar el diálogo (por ejemplo, porque la tapa del ordenador portátil está cerrada), se utiliza la opción DenySuspend para determinar si se acepta o se rechaza la solicitud de suspensión.

Multi Archivos de configuración de gestores de colas, qm.ini

Un archivo de configuración del gestor de colas, `qm.ini`, contiene información relevante para un gestor de colas específico. Atributos que se pueden utilizar para modificar la configuración de un gestor de colas individual y sustituir lo valores de IBM MQ.

Hay un archivo de configuración de gestor de colas para cada gestor de colas. El archivo `qm.ini` se crea automáticamente cuando se crea el gestor de colas con el que está asociado.

Nota: Para obtener más información sobre cómo y cuándo editar un archivo `qm.ini` y sobre cuándo entran en vigor los cambios que realice en el archivo, consulte “Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms” en la página 95.

El mandato **strmqm** comprueba la sintaxis de las stanzas CHANNELS y SSL en el archivo `qm.ini` antes de iniciar completamente el gestor de colas, lo que hace que sea mucho más fácil ver lo que está mal, y corregirlo rápidamente si **strmqm** detecta que el archivo `qm.ini` contiene errores. Si desea más información, consulte strmqm.

Ubicación de los archivos qm.ini

Linux AIX En sistemas AIX and Linux, se mantiene un archivo `qm.ini` en la raíz del árbol de directorios que ocupa el gestor de colas. Por ejemplo, la vía de acceso y el nombre de un archivo de configuración para un gestor de colas llamado QMNAME es:

```
/var/mqm/qmgrs/QMNAME/qm.ini
```

Windows En sistemas Windows, la ubicación del archivo qm.ini está determinada por el valor WorkPath, especificado en la clave HKLM\SOFTWARE\IBM\WebSphere MQ. Por ejemplo, la ruta y el nombre de un archivo de configuración de un gestor de colas llamado QMNAME son estos:

```
C:\ProgramData \IBM \MQ\qmgrs\QMNAME\qm.ini
```

IBM i Se mantiene un archivo qm.ini en *mqmdata directory*/*QMNAME*/qm.ini, donde *mqmdata directory* es /QIBM/UserData/mqm de forma predeterminada y *QMNAME* es el nombre del gestor de colas al que se aplica el archivo de inicialización.

Nota: Puede cambiar *mqmdata directory* en el archivo mqs.ini.

El nombre del gestor de colas puede tener una longitud de hasta 48 caracteres. Sin embargo, esto no garantiza que el nombre sea válido o exclusivo. Por lo tanto, se generará un nombre de directorio basado en el nombre del gestor de colas. Este proceso es conocido como *transformación de nombres*. Para obtener una descripción, consulte IBM MQ nombres de archivo y Nombres de objeto en IBM i.

qm.ini stanzas

Atención:

- Este tema enlaza con más información sobre las stanzas del archivo qm.ini . Cada stanza contiene información sobre los parámetros de dicha stanza, incluido un ejemplo cuando sea apropiado.
- Cada stanza muestra la plataforma, o plataformas, de IBM MQ for Multiplatforms a la que se aplica dicha stanza.

Multi Configuración automática de qm.ini en el inicio

Puede configurar el gestor de colas para aplicar automáticamente el contenido de un archivo, o conjunto de archivos, que contiene alteraciones temporales de qm.ini , en cada inicio de gestor de colas.

Puede utilizar esto para tener una configuración que se puede modificar y reproducir automáticamente en el siguiente reinicio del gestor de colas. Por ejemplo, si las alteraciones temporales de qm.ini se encuentran en una unidad montada, es posible tener una configuración centralizada en la que se aplica la última versión a cada gestor de colas a medida que se inician.

Puede utilizar esta funcionalidad para simplificar la creación de un clúster uniforme, utilizando la funcionalidad automática del clúster. Para obtener un ejemplo, consulte “Creación de un nuevo clúster uniforme” en la página 448.

Nota: Estas alteraciones temporales solo se aplican en el inicio del gestor de colas y no pueden influir en la creación del gestor de colas. Por ejemplo, no puede establecer el número de archivos de registro primarios con esta función.

Antes de empezar

Puede utilizar:

1. Un solo archivo y crear un archivo de texto que contenga cambios en el archivo qm.ini.
2. Un conjunto de archivos de formato qm.ini:
   - Para identificar un directorio en el que existirán las configuraciones
   - En este directorio, cree archivos, cada uno con la extensión .ini, por ejemplo, qminisettings.ini.

El archivo o archivos sólo necesitan contener la stanza y los valores **attribute**=*value* para los elementos que cambian. Por ejemplo, para actualizar el atributo **MaxChannels** de la stanza Canales, el archivo podría contener:

```
Channels:
 MaxChannels=1234
```

Tenga en cuenta que en los archivos de alteración temporal `qm.ini`, las líneas con el prefijo # se tratan como un comentario.

Habilitación de la configuración automática de atributos de archivo qm.ini

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

Puede configurar un gestor de colas existente para habilitar la configuración MQSC automática, añadiendo el **AutoConfig** atributo de stanza **IniConfig**, apuntando a un archivo o directorio válido. Por ejemplo:

```
AutoConfig:
   IniConfig=C:\MQ_Configuration\uniclus.ini
```

¿Cómo funciona el trabajo de configuración automática?

Durante el inicio del gestor de colas, se valida la configuración identificada por el atributo de stanza AutoConfig **IniConfig**, para garantizar una sintaxis válida y, a continuación, se almacena en el árbol de datos de gestor de colas en el directorio `autocfg` como un único archivo `cached.ini`.

Cuando se procesan varios archivos de un directorio, se procesan en orden alfabético.

Durante el primer inicio del gestor de colas, la imposibilidad de leer el archivo o el directorio impide que se inicie el gestor de colas, con un mensaje de error apropiado, tanto en la consola como en el registro de errores del gestor de colas.

En los reinicios posteriores, si el archivo o directorio apunta a ser ilegible, se utiliza el archivo almacenado previamente en memoria caché y un mensaje escrito en el registro de errores del gestor de colas resalta esta información.

Cuando se utiliza el mandato **strmqm**, se aplica el contenido del archivo `cached.ini` al archivo `qm.ini` como alteraciones temporales antes de que se invoque el gestor de colas.

Esto significa que para un gestor de colas en espera, los valores se leen cuando se procesa el mandato **strmqm**, no cuando el gestor de colas pasa a estar activo.

¿Cómo se crea el archivo qm.ini de sustitución?

La primera vez que se configura la configuración de inicialización automática y se inicia el gestor de colas, se copia una copia del archivo `qm.ini` actual en el subdirectorio `autoconfig` dentro del directorio de datos del gestor de colas como `base_qm.ini`. Esto se considera la línea base desde aquí en adelante.

En cada inicio de gestor de colas, es decir, la hora de **strmqm**, el archivo `qm.ini` activo actualmente se descarta y se sustituye por una copia de `base_qm.ini`. A continuación, la configuración del archivo `cached.ini` se aplica a este archivo.

Una vez que un gestor de colas está bajo control de configuración automática, todos los cambios en el archivo `qm.ini` se deben realizar a través del archivo, o archivos, a los que se apunta utilizando el atributo **IniConfig** de la stanza AutoConfig.

Puesto que se elimina el archivo `qm.ini` existente en el inicio del gestor de colas, solo la configuración del archivo `qm.ini` proporcionado que utiliza el atributo **IniConfig** se aplica a la línea base del gestor de colas.

Si se ha modificado una stanza o un atributo a través de la configuración de inicialización automática en inicios anteriores del gestor de colas, estos cambios se eliminan, a menos que todavía estén identificados en el archivo o archivos identificados por el atributo **IniConfig**.

Debido a la recreación del archivo `qm.ini` en el inicio del gestor de colas, esto significa que se pierden los cambios manuales en el archivo `qm.ini`. Si realmente necesita realizar un cambio persistente y no puede utilizar el atributo **IniConfig** para realizar dicho cambio, puede realizar una de las acciones siguientes:

- Realice el cambio en el propio archivo `base_qm.ini`.
- Suprima el archivo `base_qm.ini`.

  Si suprime este archivo, el `base_qm.ini` se vuelve a crear en el siguiente inicio del gestor de colas, basándose en el contenido actual del archivo `qm.ini`. Esto *guarda* todos los cambios actuales a medida que se inicia la nueva línea base para el futuro.

Conceptos relacionados

"Resumen de stanzas y atributos del archivo qm.ini" en la página 112
Un resumen de los atributos de las stanzas del archivo de configuración del gestor de colas, `qmi.ini`, con enlaces a más información.

Multi Resumen de stanzas y atributos del archivo qm.ini

Un resumen de los atributos de las stanzas del archivo de configuración del gestor de colas, `qmi.ini`, con enlaces a más información.

Tabla 12. Stanzas del archivo qm.ini

Stanza y atributos	Descripción de atributos
Windows **Stanza AccessMode**	
Windows grupo de acceso [1]	Un grupo de seguridad de Windows, los miembros del cual tendrán un acceso completo a todos los archivos de datos del gestor de colas.
sección ApiExitLocal	
Nombre	El nombre que describe la salida de API que se ha pasado en el campo ExitInfoName de la estructura MQAXP.
Función	El nombre del punto de entrada de la función al módulo que contiene el código de la salida de API.
Módulo	El módulo que contiene el código de la salida de API.
Datos	Los datos que se han de pasar a la salida de API en el campo ExitData de la estructura MQAXP.
Sequence	La secuencia en que se llama a esta salida de API es relativa para las otras salidas de API.
Linux V 9.4.0 AIX **StanzaAuthToken**	
KeyStore	Vía de acceso de archivo para el almacén de claves que contiene los certificados de clave pública o las claves simétricas del emisor de confianza.
KeyStorePwdFile	Vía de acceso del archivo que contiene la contraseña cifrada para el almacén de claves.
CertLabel	La etiqueta de certificado para un certificado de clave pública o clave simétrica en el almacén de claves que se utiliza para validar las señales de autenticación.

Tabla 12. Stanzas del archivo qm.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
UserClaim	Reclame dentro de la señal que contiene información de identidad de usuario que el gestor de colas puede adoptar para comprobaciones de autorización.
AllowOSGroups	Este atributo determina si la pertenencia al grupo para el usuario adoptado está seleccionada o no.
Stanza AutoCluster	
Type	El tipo de clúster automático. La única opción válida es Uniform, que representa un clúster uniforme.
ClusterName	El nombre del clúster automático.
RepositoryName1	Nombre de gestor de colas para el primer repositorio completo del clúster automático.
Repository1Conname	El valor de nombre de conexión (CONNAME) para la forma en que los miembros del clúster automático deben conectarse al gestor de colas.
RepositoryName2	Nombre de gestor de colas para el segundo repositorio completo del clúster automático.
Repository2Conname	El valor de nombre de conexión (CONNAME) para la forma en que los miembros del clúster automático deben conectarse al gestor de colas.
Stanza AutoConfig	
MQSCConfig	Una vía de acceso de archivo completa o una vía de acceso a un directorio, donde todos los archivos `*.mqsc` se aplican al gestor de colas en cada inicio del gestor de colas.
IniConfig	Una vía de acceso de archivo completa o una vía de acceso a un directorio, donde todos los archivos `*.ini` se aplican al archivo `qm.ini` en cada inicio del gestor de colas.
stanza Channels	
MaxChannels	El número máximo de canales actuales permitidos.
MaxActiveChannels	El número máximo de canales que pueden estar activos en cualquier momento.
MaxInitiators	El número máximo de iniciadores.
MQIBindType	El enlace de las aplicaciones.
PipeLineLength	El número máximo de hebras simultáneas que utilizará un canal.
AdoptNewMCA	Qué tipos de canales pueden tener la instancia de canal existente detenida para que se pueda iniciar una nueva instancia de canal cuando IBM MQ recibe una solicitud para iniciar un canal, pero descubre que una instancia del canal ya se está ejecutando.

Tabla 12. Stanzas del archivo qm.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
AdoptNewMCATimeout	La cantidad de tiempo, en segundos, que la nueva instancia de canal deberá esperar a que finalice la instancia de canal anterior.
AdoptNewMCACheck	El tipo de comprobación necesario cuando se habilita el atributo **AdoptNewMCA**.
ChlauthEarlyAdopt	El orden en el que se procesan las reglas de autenticación de conexión y de autenticación de canal.
PasswordProtection	Si las credenciales especificadas por una aplicación deben protegerse con la protección de contraseña MQCSP, si el canal no utiliza el cifrado TLS.
IgnoreSeqNumberMismatch	Controla la forma en que el gestor de colas maneja la no coincidencia de número de secuencia durante el inicio del canal.
Stanza Connection	
DefaultBindType	Indica si las aplicaciones y el gestor de colas, que se ejecutan en procesos separados, comparten algunos recursos o no comparten ninguno.
Stanza DiagnosticMessages	
name	Nombre de una stanza.
Servicio	Un servicio que está siendo habilitado por esta stanza.
ExcludeMessage	Los mensajes que no se van a escribir en el registro de errores del gestor de colas.
SuppressMessage	Los mensajes que se van a escribir en el registro de errores del gestor de colas solo una vez en un intervalo de tiempo especificado.
IBM i SuppressInterval	El intervalo de tiempo, en segundos, durante el cual los mensajes especificados en **SuppressMessage** se escriben solo una vez en el registro de errores del gestor de colas.
Severities	Una lista de los niveles de gravedad separados por comas.
FilePath	La vía de acceso donde se van a escribir los archivos de registro. (Solo está soportado cuando el atributo Service está establecido en File.)
FilePrefix	Prefijo de los archivos de registro. (Solo está soportado cuando el atributo Service está establecido en File.)
FileSize	El tamaño con el que se rota el registro. (Solo está soportado cuando el atributo Service está establecido en File.)
Formato	El formato del archivo. (Solo está soportado cuando el atributo Service está establecido en File.)
Linux AIX Syslog	El servicio Syslog que envía los mensajes sin filtrar al syslog utilizando la especificación de mensajes de diagnóstico de formato JSON.

Tabla 12. Stanzas del archivo qm.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
Linux AIX Ident	El valor ident asociado a las entradas del syslog. (Solo está soportado cuando el atributo Service está establecido en Syslog.)
stanza ExitPath	
ExitsDefaultPath	La vía de acceso de los programas de salida de usuario en el sistema del gestor de colas (32-bits).
ExitsDefaultPath64	La vía de acceso de los programas de salida de usuario en el sistema de gestor de colas (64-bits).
Stanza ExitPropertiesLocal	
CLWLMode	Indica si la salida de carga de trabajo del clúster (CLWL) se ejecuta en la modalidad FAST o en la modalidad SAFE.
IBM i Linux AIX **Stanza Filesystem**	
IBM i Linux AIX ValidateAuth	Permita a los usuarios que no son miembros del grupo `mqm` acceder a directorios y archivos de errores.
stanza Log	
LogPrimaryFiles	Los archivos de anotaciones asignados cuando se crea el gestor de colas.
LogSecondaryFiles	Los archivos de anotaciones que se asignan cuando se agotan los archivos primarios.
LogFilePages	El número de páginas de archivo de registro. (El tamaño del archivo de registro se especifica en unidad de páginas de 4 KB.)
LogType	El tipo de registro que va a utilizar el gestor de colas (circular o lineal).
LogBufferPages	La cantidad de memoria asignada a los registros de almacenamiento intermedio para grabación, especificando el tamaño de los almacenamientos intermedios en unidades de páginas de 4 KB.
LogPath	El directorio en el que residen los archivos de registro de un gestor de colas.
LogWriteIntegrity	El método que utiliza el registrador de anotaciones para grabar los registros de anotaciones de forma fiable.
LogManagement	El método utilizado para gestionar las extensiones de registro, ya sea manualmente o mediante el gestor de colas.
Windows **Stanza LU62**	
Windows TPName	El nombre de TP que debe iniciarse en la ubicación remota.
Windows Library1	El nombre de la DLL de APPC.

Tabla 12. Stanzas del archivo qm.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
Windows Library2	Igual que Library1, utilizada si el código se almacena en dos bibliotecas distintas.
CP4I **Stanza NativeHAInstance**	
"Nombre" en la página 149	El nombre de instancia que se utilizó cuando se creó la instancia del gestor de colas.
"ReplicationAddress" en la página 149	El nombre de host, la dirección en formato decimal con puntos IPv4 o hexadecimal IPv6 de la instancia.
CP4I **Stanza NativeHALocalInstance**	
"LocalName" en la página 150	El nombre de la stanza NativeHALocalInstance, tomada del nombre de instancia de réplica de registro especificado cuando se crea el gestor de colas HA nativo.
"KeyRepository" en la página 150	La ubicación del repositorio de claves que contiene el certificado digital que se debe utilizar para la protección del tráfico de réplica de registro.
"CertificateLabel" en la página 151	Etiqueta de certificado que identifica el certificado digital que se debe utilizar para la protección del tráfico de réplica de registro.
"CipherSpec" en la página 151	La CipherSpec de MQ que se utilizará para proteger el tráfico de réplica de registro.
"LocalAddress" en la página 151	La dirección de la interfaz de red local que acepta el tráfico de réplica de registro.
"HeartbeatInterval" en la página 151	El intervalo de latidos define la frecuencia en milisegundos a la que una instancia activa de un gestor de colas de HA nativo envía una pulsación de red.
"HeartbeatTimeout" en la página 151	El tiempo de espera de latido define cuánto tiempo espera una instancia de réplica de un gestor de colas de HA nativo antes de decidir que la instancia activa no responde.
"RetryInterval" en la página 151	El intervalo de reintento define la frecuencia en milisegundos a la que un gestor de colas HA nativo debe reintentar un enlace de réplica anómalo.
Windows **Stanza NETBIOS**	
Windows LocalName	El nombre por el que se conoce a la máquina en la LAN.
Windows AdapterNum	El número del adaptador de la LAN.
Windows NumSess	El número de sesiones que se debe asignar.
Windows NumCmds	El número de mandatos que se debe asignar.
Windows NumNames	El número de nombres que se debe asignar.
Windows Library1	El nombre de la DLL de NetBIOS.

Tabla 12. Stanzas del archivo qm.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
Stanza QMErrorLog	
ErrorLogTamaño	Especifica el tamaño de las anotaciones cronológicas de errores del gestor de colas que se incluye en la copia de seguridad.
ExcludeMessage	Especifica mensajes que no se deben grabar en el registro de errores del gestor de colas.
SuppressMessage	Especifica que se graben mensajes en el registro de errores del gestor de colas sólo una vez en un intervalo de tiempo especificado.
SuppressInterval	Especifica el intervalo de tiempo, en segundos, durante el cual los mensajes especificados en SuppressMessage se escriben solo una vez en el registro de errores del gestor de colas.
Linux AIX **Stanza de modalidad restringida** [2]	
Linux AIX ApplicationGroup	El nombre de la cola de transmisión local donde se transfieren los mensajes remotos si no se define explícitamente una cola de transmisión como destino.
Stanza Security	
ClusterQueueAccessControl	Compruebe el control de accesos de las colas de clúster o las colas totalmente calificadas alojadas en los gestores de colas del clúster.
Windows GroupModel	Indica si el Gestor de autorizaciones sobre objetos (OAM) comprueba los grupos globales al determinar la pertenencia a grupos de un usuario en Windows.
sección Service	
Nombre	El nombre del servicio necesario.
EntryPoints	El número de puntos de entrada definidos para el servicio.
Windows SecurityPolicy	En Windows, la política de seguridad para cada gestor de colas
Linux AIX SecurityPolicy	En AIX and Linux, indica si el gestor de colas utiliza la autorización basada en usuario o basada en grupo. A partir de IBM MQ 9.3.0, también puede crear un nombre de usuario que no sea del sistema operativo.
SharedBindingsUserId	Solo para los enlaces compartidos, indica si el campo UserIdentifier en la estructura IdentityContext, de la función MQZ_AUTHENTICATE_USER, es el ID de usuario efectivo o el ID de usuario real.
FastpathBindingsUserId	Solo para los enlaces de vía de acceso rápida, indica si el campo UserIdentifier en la estructura IdentityContext, de la función MQZ_AUTHENTICATE_USER, es el ID de usuario efectivo o el ID de usuario real.

Tabla 12. Stanzas del archivo qm.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
IsolatedBindingsUserId	Solo para los enlaces aislados, indica si el campo UserIdentifier en la estructura IdentityContext, de la función MQZ_AUTHENTICATE_USER, es el ID de usuario efectivo o el ID de usuario real.
sección ServiceComponent	
Servicio	El nombre del servicio necesario.
Nombre	El nombre descriptivo del componente de servicio.
Módulo	El nombre del módulo que contendrá el código para este componente.
ComponentDataSize	El tamaño, en bytes, del área de datos del componente que se pasa al componente en cada llamada.
Windows **Stanza SPX**	
Windows Socket	Número de socket de SPX en notación hexadecimal.
Windows BoardNum	El número de adaptador de la LAN.
Windows KeepAlive	Permite activar o desactivar la función KeepAlive.
Windows Library1	Nombre DLL de SPX.
Windows Library2	Es el mismo que el valor de LibraryName1 y se utiliza si el código se almacena en dos bibliotecas distintas.
Windows ListenerBacklog	Permite alterar temporalmente el número predeterminado de solicitudes pendientes para el escucha de SPX.
Stanza SSL	
OutboundSNI	Especifica si los clientes que tengan habilitado SNI establecerán SNI en el nombre de canal IBM MQ de destino en el sistema remoto al iniciar una conexión TLS o al nombre de host.
AllowOutboundSNI	Especifica si los clientes que tengan habilitado SNI establecerán SNI en el nombre de canal IBM MQ de destino en el sistema remoto al iniciar una conexión TLS. **Atención:** Deprecated A partir de IBM MQ 9.3.0 , esta propiedad está en desuso. En su lugar, utilice **OutboundSNI** .
AllowedCipherSpecs	Especifica una lista personalizada de CipherSpecs que están ordenadas y habilitadas para ser utilizadas con canales IBM MQ en multiplataformas.
AllowTLSV13	Indica si un gestor de colas puede utilizar las TLS 1.3 CipherSpecs.
CDPCheckExtensions	Indica si los canales TLS de este gestor de colas intentan comprobar los servidores CDP que se especifican en extensiones de certificado CrlDistributionPoint.

Tabla 12. Stanzas del archivo qm.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
MinimumRSAKeyTamaño	Especifica el tamaño de clave mínimo que deben tener los certificados RSA para poder aceptarse.
OCSPAuthentication	La acción que se va a llevar a cabo cuando no se puede determinar un estado de revocación desde un servidor OCSP.
OCSPCheckExtensions	Indica si los canales TLS de este gestor de colas intentan comprobar los servidores OCSP que se especifican en extensiones de certificado AuthorityInfoAccess.
OCSPTimeout	El número de segundos que se debe esperar un programa de respuesta OCSP al realizar una comprobación de revocación.
ALW PeerCertChainValidation	El valor de validación de certificado de IBM Global Security Kit (GSKit).
SSLHTTPProxyName	El nombre de host o la dirección de red del servidor proxy HTTP que GSKit va a utilizar para las comprobaciones de OCSP.
SSLHTTPConnectTimeout	El número de segundos que se va a esperar a que una conexión de red se establezca correctamente en un servidor HTTP al realizar una comprobación de revocación.
stanza Subpool “3” en la página 121	Esta stanza la crea IBM MQ. No la cambie.
ShortSubpoolName “3” en la página 121	Un nombre que corresponde a un directorio y un enlace simbólico creado dentro del directorio /var/mqm/sockets, que utiliza IBM MQ para las comunicaciones internas entre sus procesos en ejecución.
Windows **Stanza TCP**	
Puerto	El número de puerto predeterminado, en notación decimal, para sesiones TCP/IP.
Windows Library1	El nombre de la DLL de TCP/IP.
KeepAlive	Permite activar o desactivar la función KeepAlive.
ListenerBacklog	Permite alterar temporalmente el número predeterminado de peticiones pendientes para el escucha de TCP/IP.
Connect_Timeout	El número de segundos antes de que un intento de conectar el socket sobrepase el tiempo de espera.
SndBuffSize	El tamaño en bytes del almacenamiento intermedio de envío TCP/IP que utiliza el extremo emisor de los canales.
RcvBuffSize	El tamaño en bytes del almacenamiento intermedio de recepción TCP/IP que utiliza el extremo receptor de los canales.
RcvSndBuffSize	Tamaño en bytes del almacenamiento intermedio de envío TCP/IP que utiliza el extremo emisor de un canal receptor.

Tabla 12. Stanzas del archivo qm.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
RcvRcvBuffSize	Tamaño en bytes del almacenamiento intermedio de recepción TCP/IP que utiliza el extremo receptor de un canal receptor.
SvrSndBuffSize	Tamaño en bytes del almacenamiento intermedio de envío TCP/IP utilizado por el extremo de servidor de un canal de conexión de cliente y de servidor.
SvrRcvBuffSize	Tamaño en bytes del almacenamiento intermedio de recepción TCP/IP utilizado por el extremo de servidor del canal de conexión de cliente y de servidor.
Multi SecureCommsOnly	Especifica si se permite la comunicación de texto sin formato, el valor predeterminado o no se permite.
Stanza de parámetros de ajuste	
SuppressDspAuthFail	Si el gestor de colas suprime la generación de sucesos de autorización y la escritura de mensajes de error AMQ8077 en el registro de errores cuando falla una comprobación de autorización, si la conexión carece de autorización +dsp sobre un objeto.
ImplSyncOpenOutput	El número mínimo de aplicaciones que tienen la cola abierta para la colocación, antes de que se pueda habilitar un punto de sincronización implícito para una colocación persistente, fuera del punto de sincronización.
UniformClusterNombre	El nombre del clúster de IBM MQ que está utilizando como un clúster uniforme.
OAMLdapConnectTiempo de espera	El tiempo máximo, en segundos, que el cliente LDAP esperará para establecer una conexión TCP con el servidor.
OAMLdapQueryTimeLimit	El tiempo máximo, en segundos, que el cliente LDAP esperará para recibir una respuesta a una solicitud LDAP del servidor.
OAMLdapResponseWarningTime	Si una conexión con un servidor LDAP ha tardado más tiempo que el número de segundos de umbral especificado por el parámetro **OAMLdapResponseWarningTime** , se grabará un mensaje AMQ5544W en el registro de errores.
ExpiryInterval	Indica la frecuencia con la que el gestor de colas explora las colas en busca de mensajes caducados que otras actividades de cola todavía no han limpiado. Es un intervalo de tiempo en segundos.
LivenessHeartBeatLen	Configura la frecuencia con la que el gestor de colas comprueba que las grabaciones en el registro se realizan a una velocidad razonable.
ECHeartBeatLen	Configura la frecuencia de las comprobaciones de estado generales del gestor de colas.

Tabla 12. Stanzas del archivo qm.ini (continuación)	
Stanza y atributos	**Descripción de atributos**
FileLockHeartBeatLen	Cambia el valor predeterminado para las comprobaciones de bloqueo de archivo para un gestor de colas de varias instancias que el controlador de ejecución realiza periódicamente para asegurarse de que todavía mantiene el bloqueo exclusivo en el archivo de varias instancias primario.
Stanza Variables	
atributo=valor	Un nombre y un valor asociado para utilizarlo como una inserción durante las definiciones de mandato de script de IBM MQ.
sección XAResourceManager	
Nombre	La instancia del gestor de recursos.
SwitchFile	El nombre completo del archivo de carga que contiene la estructura de conmutación XA del gestor de recursos.
XAOpenString	La serie de datos que se ha de pasar al punto de entrada xa_open del gestor de recursos.
XACloseString	La serie de datos que se ha de pasar al punto de entrada xa_close del gestor de recursos.
ThreadOfControl	El valor que utiliza el gestor de colas para la serialización cuando necesita llamar al gestor de recursos desde uno de sus propios procesos multihebra. Obligatorio para Windows.

Notas:

1. La stanza AccessMode establece la opción **-a [r]** en el mandato **crtmqm**. No modifique la stanza AccessMode después de haber creado el gestor de colas.
2. La stanza RestrictedMode se establece mediante la opción **-g** en el mandato **crtmqm**. No modifique esta stanza después de haber creado el gestor de colas. Si no utiliza la opción **-g** , la stanza no se crea en el archivo `qm.ini` .
3. La stanza Subpool, y el atributo ShortSubpoolName dentro de dicha stanza, se escriben automáticamente mediante IBM MQ, cuando se crea un gestor de colas. IBM MQ elige un valor para ShortSubpoolName. No modifique dicho valor.

Windows Stanza AccessMode del archivo qm.ini

La modalidad de acceso solo se aplica a servidores Windows. La stanza AccessMode del archivo qm.ini se establece mediante la opción `-a [r]` en el mandato **crtmqm**. No modifique la stanza AccessMode después de haber creado el gestor de colas.

Utilice la opción de grupo de acceso (`-a [r]`) del mandato **crtmqm** para especificar un grupo de seguridad de Windows, los miembros del cual obtendrán acceso completo a todos los archivos de datos del gestor de colas. El grupo puede ser un grupo local o global, dependiendo de la sintaxis utilizada. La sintaxis válida para el nombre de grupo es la siguiente:

LocalGroup
Nombre de dominio \ *Nombre de grupo local*
Nombre de grupo global@*Nombre de dominio*

Debe definir el grupo adicional antes de ejecutar el mandato **crtmqm** con la opción `-a [r]`.

Si especifica el grupo utilizando `-ar` en lugar de `-a`, el grupo mqm local no obtiene acceso a los archivos de datos del gestor de colas. Utilice esta opción si el sistema de archivos que contiene los archivos de datos del gestor de colas no da soporte a entradas de control de acceso para grupos definidos localmente.

El grupo normalmente es un grupo de seguridad global, que se utiliza para proporcionar a los gestores de colas multiinstancia acceso a datos de un gestor de colas compartido y a una carpeta de registros. Utilice el grupo de acceso de seguridad adicional para establecer permisos de lectura y escritura en la carpeta o para compartir los archivos de registro y datos de gestor de colas que contiene.

El grupo de acceso de seguridad adicional es una alternativa al uso del grupo local denominado `mqm` para establecer permisos en la carpeta que contiene los datos y los registros del gestor de colas. A diferencia del grupo local `mqm`, puede hacer que el grupo de acceso de seguridad adicional sea un grupo local o global. Debe ser un grupo global para establecer permisos en las carpetas compartidas que contengan archivos de datos y registros utilizados por gestores de colas multiinstancia.

El sistema operativo Windows comprueba los permisos de acceso para leer y escribir archivos de registro y datos de gestor de colas. Comprueba los permisos del ID de usuario que está ejecutando los procesos del gestor de colas. El ID de usuario que se comprueba depende de si ha iniciado el gestor de colas como servicio o lo ha iniciado de forma interactiva. Si ha iniciado el gestor de colas como servicio, el ID de usuario comprobado por el sistema Windows es el ID de usuario configurado con el asistente **Preparar** IBM MQ. Si ha iniciado el gestor de colas de forma interactiva, el ID de usuario comprobado por el sistema Windows es el ID de usuario que ha ejecutado el mandato **strmqm**.

El ID de usuario debe ser miembro del grupo `mqm` local para iniciar el gestor de colas. Si el ID de usuario es miembro del grupo de acceso de seguridad adicional, el gestor de colas puede leer y escribir archivos a los que se les otorga permisos utilizando el grupo.

Restricción: Puede especificar un grupo de acceso de seguridad adicional sólo en el sistema operativo Windows. Si especifica un grupo de acceso de seguridad adicional en otros sistemas operativos, el mandato **crtmqm** devuelve un error.

Stanza de ejemplo

```
AccessMode:
SecurityGroup=wmq\wmq
```

Conceptos relacionados

“Proteger directorios y archivos de datos y registros del gestor de colas no compartidos en Windows” en la página 564

“Proteger directorios y archivos de datos y registros compartidos del gestor de colas en Windows” en la página 561

Tareas relacionadas

“Creación de gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio en Windows” en la página 536

Referencia relacionada

crtmqm (crear gestor de colas)

Multi Stanza ApiExitLocal del archivo qm.ini

La stanza local ApiExitespecifica rutinas de salida de API para un gestor de colas.

Para un servidor, modifique la stanza local ApiExitdel archivo `qm.ini` para identificar las rutinas de salida de API para un gestor de colas.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades del gestor de colas de IBM MQ Explorer `Exits` .

Para un cliente, modifique la stanza ApiExitlocal en el archivo `mqclient.ini` para identificar las rutinas de salida de API para un gestor de colas.

Visión general

La stanza `ApiExitLocal` solo permite la especificación de un único `Module`, aunque deben proporcionarse cuatro módulos, como se indica a continuación:

- 32 bits sin hebras
- 32 bits con hebras
- 64 bits sin hebras
- 64 bits con hebras

Tenga en cuenta que IBM MQ añade `_r` al nombre de módulo proporcionado para identificar la versión con hebras de la salida, pero IBM MQ no proporciona un mecanismo directamente equivalente para las variantes de 32 bits y 64 bits.

Las versiones de `amqsaxe0` y `amqsaxe0_r` que se suministran en *prefix*`/mqm/samp/bin` se crean para el tamaño nativo del gestor de colas en la plataforma para la que se crean (ahora todos de 64 bits) y solo pueden utilizarse en las aplicaciones que se ejecutan con el mismo tamaño nativo.

Si se proporciona un nombre de módulo no calificado, IBM MQ busca `/var/mqm/exits` para las variantes de 32 bits y en `/var/mqm/exits64` para las variantes de 64 bits

Por ejemplo, `module=amqsaxe` significa:

```
/var/mqm/exits/amqsaxe  - 32 bit unthreaded variant
/var/mqm/exits/amqsaxe_r - 32 bit threaded variant
/var/mqm/exits64/amqsaxe - 64 bit unthreaded variant
/var/mqm/exits64/amqsaxe_r - 64 bit threaded variant
```

Windows En sistemas Windows, también puede utilizar el mandato **amqmdain** para cambiar las entradas para salidas de API. (Para identificar rutinas de salida de API para todos los gestores de colas, utilice las stanzas ApiExitCommon y ApiExitTemplate, tal como se describe en "Stanzas ApiExitCommon y ApiExitTemplate del archivo mqs.ini" en la página 103.)

No olvide que para que la salida de API funcione correctamente, el mensaje del servidor debe enviarse al cliente sin convertir. Después de que se haya procesado la salida de la API, el mensaje debe convertirse en el cliente. Por consiguiente, esto requiere que haya instalado todas las salidas de conversión en el cliente.

Para obtener más información sobre cómo utilizar estos atributos, consulte Configuración de salidas de API.

Parámetros

Name=nombre_ApiExit
El nombre que describe la salida de API que se ha pasado en el campo ExitInfoName de la estructura MQAXP.

Este nombre debe ser exclusivo, con un máximo de 48 caracteres de longitud y sólo puede contener caracteres válidos para los nombres de objetos IBM MQ (por ejemplo, nombres de colas).

Function=nombre_función
El nombre del punto de entrada de la función al módulo que contiene el código de la salida de API. Este punto de entrada es la función MQ_INIT_EXIT.

La longitud de este campo está limitada a MQ_EXIT_NAME_LENGTH.

Module=nombre_módulo
El módulo que contiene el código de la salida de API.

Si este campo contiene el nombre de vía de acceso completo del módulo, se utilizará tal y como está. Si este campo contiene solo el nombre de módulo, el módulo se encuentra utilizando el atributo **ExitsDefaultPath** en la stanza `ExitPath` del archivo `qm.ini`.

En plataformas que dan soporte a bibliotecas con hebras independientes, debe proporcionar tanto una versión sin hebras como una versión con hebras del módulo de salida de API. La versión con hebras debe tener un sufijo `_r`. La versión con hebras del apéndice de aplicación de IBM MQ añade implícitamente un sufijo `_r` al nombre de módulo especificado antes de cargarlo.

La longitud de este campo está limitada a la longitud máxima de vía de acceso a la que dé soporte la plataforma.

Data=nombre_datos

Los datos que se han de pasar a la salida de API en el campo ExitData de la estructura MQAXP.

Si incluye este atributo, se suprimirán los espacios en blanco iniciales y de cola, la serie restante se truncará a 32 caracteres y el resultado se pasará a la salida. Si omite este atributo, se pasará el valor predeterminado de 32 espacios en blanco.

La longitud máxima de este campo es de 32 caracteres.

Sequence=número_secuencia

La secuencia en que se llama a esta salida de API es relativa para las otras salidas de API. Se llama antes a una salida con un número de secuencia bajo que a una salida con un número de secuencia más alto. No es necesario que los números de secuencia de las salidas sean contiguos. Una secuencia de 1, 2, 3 tiene el mismo resultado que una secuencia de 7, 42, 1096. Si dos salidas tienen el mismo número de secuencia, el gestor de colas decide a cuál de ellos llamará en primer lugar. Puede saber a cuál se ha llamado después del suceso, colocando la hora o un marcador en ExitChainArea, que se indica mediante ExitChainAreaPtr en MQAXP, o escribiendo su propio archivo de anotaciones.

Este atributo es un valor numérico sin signo.

Stanza de ejemplo

```
ApiExitLocal:
   Name=ClientApplicationAPIchecker
   Sequence=3
   Function=EntryPoint
   Module=/usr/Dev/ClientAppChecker
   Data=9.20.176.20
```

Linux V 9.4.0 AIX Stanza AuthToken del archivo `qm.ini`

Utilice la stanza **AuthToken** para configurar el gestor de colas para validar las señales de autenticación proporcionadas por las aplicaciones de conexión. Si el servicio de autenticación da soporte a un punto final JWKS para la configuración de claves, normalmente es una opción preferible.

Consulte Configuración de un gestor de colas para aceptar señales de autenticación utilizando un punto final JWKS para obtener más información.

La stanza AuthToken

KeyStore= serie

Vía de acceso de archivo para el almacén de claves que contiene los certificados de clave pública y las claves simétricas del emisor de confianza. Puede añadir las claves a un almacén de claves existente o crear un almacén de claves nuevo. Para obtener más información, consulte Configuración de un gestor de colas para aceptar señales de autenticación. El gestor de colas utiliza las claves del almacén de claves para verificar que la señal de autenticación que presenta la aplicación está firmada por el emisor de confianza.

Puede utilizar un almacén de claves CMS con la extensión de archivo `.kdb` o un almacén de claves PKCS#12 con la extensión de archivo `.p12`. Si el archivo de almacén de claves no existe o no se puede acceder a él, se genera un error `AMQ7076E: Valor no válido para atributo en archivo ini` en el registro de errores del gestor de colas.

Asegúrese de que el tipo de almacén de claves coincide con la extensión de nombre de archivo para el almacén de claves. IBM MQ detecta el formato correcto del almacén de claves; sin embargo, las

incoherencias pueden provocar otros problemas administrativos si el tipo de almacén de claves y la extensión de nombre de archivo no coinciden.

La longitud máxima de la vía de acceso del archivo de almacén de claves es de 256 caracteres.

KeyStorePwdFile= serie

Vía de acceso del archivo que contiene la contraseña cifrada para el almacén de claves. El archivo debe contener la contraseña cifrada como una sola línea de texto. No se aceptan contraseñas de texto sin formato.

Utilice el mandato **runqmcred** para cifrar la contraseña antes de guardarla en el archivo de contraseñas del almacén de claves. El archivo de contraseña de almacén de claves sólo debe contener la contraseña cifrada que se crea ejecutando el mandato **runqmcred** .

La longitud máxima de la contraseña de texto sin formato antes de cifrarla es de 1024 caracteres.

Este parámetro es opcional. Si no se proporciona, el gestor de colas busca un archivo de ocultación con la contraseña en el mismo directorio y con el mismo nombre que el almacén de claves, pero con la extensión de archivo `.sth`. Si no se encuentra el archivo de ocultación, se rechaza la configuración y se envía el mensaje de error AMQ7006E al registro de errores del gestor de colas. Para obtener más información sobre las opciones para almacenar contraseñas de almacén de claves, consulte Cifrado de contraseñas de repositorio de claves.

La longitud máxima de la vía de acceso del archivo de contraseñas es de 256 caracteres.

CertLabel= serie

La etiqueta de certificado para un certificado de clave pública o clave simétrica en el almacén de claves que se utiliza para validar las señales de autenticación. Puede proporcionar hasta 32 etiquetas de certificado repitiendo el atributo **CertLabel** .

Cuando añada certificados al almacén de claves del gestor de colas, déles etiquetas significativas. Las etiquetas de certificado distinguen entre mayúsculas y minúsculas. Pueden contener caracteres alfanuméricos, caracteres de puntuación y espacios. Si se detecta un carácter no válido, se devuelve un error y se escribe un mensaje de error en el registro de errores de IBM MQ .

Los emisores de señales de confianza pueden proporcionar varios certificados de clave pública y claves simétricas. Por ejemplo, los certificados de clave pública tienen periodos de validez. Cuando están a punto de caducar, el emisor de señal proporciona un nuevo certificado con una nueva fecha de caducidad. Durante un tiempo, ambos certificados pueden ser válidos.

Cuando las aplicaciones presentan señales para la autenticación, la lista de **CertLabels** se comprueba hasta que se encuentra una clave válida que se ha utilizado para firmar la señal. Si se encuentra la coincidencia, se valida la firma de señal.

Si no se especifica **CertLabel** , la conexión de la aplicación que presenta la señal falla con el código de razón `2063 MQRC_SECURITY_ERROR`, y el mensaje `AMQ5786E: Error de configuración de señal de autenticación` se graba en el registro de errores del gestor de colas.

La longitud máxima de la etiqueta de certificado es de 64 caracteres.

Por ejemplo,

```
AuthToken:
   KeyStore=/var/mqm/qmgrs/qmgrs/qm1/tokenissuer/key.kdb
   KeyStorePwdFile=/var/mqm/qmgrs/qm1/tokenissuer/key.pw
   CertLabel=token
   CertLabel=rsakey
   CertLabel=mark
   ... up to 32 CertLabel fields
```

UserClaim= serie

Reclame dentro de la señal que contiene el ID de usuario que el gestor de colas adopta para las comprobaciones de autorización.

Este parámetro es opcional si el gestor de colas está configurado con **ADOPTCTX(NO)**. Si se utiliza **ADOPTCTX(YES)** , este parámetro es necesario. **ADOPTCTX** es un atributo presente en el objeto de información de autenticación (AUTHINFO) al que hace referencia el atributo **CONNAUTH** del gestor de colas.

Para adoptar una identidad, la señal debe contener una reclamación con el nombre especificado en el atributo **UserClaim** de la stanza **AuthToken** y se debe utilizar **ADOPTCTX(YES)** .

Por ejemplo, si la señal incluye una reclamación `"AppUser": "MyUserName"`, debe especificar `UserClaim=AppUser` en la stanza AuthToken del archivo `qm.ini` para adoptar la identidad `"MyUserName"` para la autorización.

La longitud máxima del valor de atributo **UserClaim** es de 128 caracteres.

Nota: DeIBM MQ 9.4.0 , Si elAuthToken se especifica la estrofa, el valor efectivo de la**SecurityPolicy** El atributo de la estrofa de servicio se establece en`UserExternal` . La autenticación de token no está disponible si**SecurityPolicy** está explícitamente establecido en `Grupo` en la estrofa de Servicio. Si**SecurityPolicy** se establece en `Grupo` , eliminar el**SecurityPolicy** atributo de la sección Servicio y, a continuación, reinicie el gestor de colas. Para más información, verSecurityPolicy .

Nota: Utilice el atributo **ADOPTCTX** del objeto de información de autenticación para controlar si el ID de usuario de la señal se adopta para las comprobaciones de autorización. Cuando crea el gestor de colas, este atributo se establece en **ADOPTCTX(YES)**. Este valor hace que se adopte el ID de usuario de la señal. El ID de usuario debe cumplir los requisitos para los ID de usuario en las señales de autenticación. Para obtener más información, consulte ID de usuario en señales de autenticación. Si la reclamación de usuario de señal contiene un ID de usuario que no cumple los requisitos, la conexión se rechaza con el código de razón **2035 MQRC_NOT_AUTHORIZED**. Si se establece **ADOPTCTX(NO)** , la señal sólo se utiliza para la autenticación y se debe utilizar otro usuario para la autorización.

AllowOSGroups=NO (valor predeterminado) |YES
El valor por omisión es NO. Determina si una identidad que se adopta desde una señal se trata como un usuario de sistema operativo (SO) y si se respetan las pertenencias a grupos del usuario de SO coincidente durante la autorización.

AllowOSGroups= NO | N
Las comprobaciones de autorización sólo se basan en el nombre del usuario que se adopta de la señal.

AllowOSGroups= SÍ | Y
Las comprobaciones de autorización se basan en el nombre del usuario y también se comprueban los grupos a los que pueden pertenecer.

Nota: El atributo **allowOSGroups** de la stanza **AuthToken** sigue teniendo efecto durante la autenticación de señal, incluso si el resto de la configuración de validación de señal se gestiona a través de la stanza **JWKS** .

Stanza de ejemplo-sólo autenticación

Es válido que esta stanza contenga únicamente el atributo **AllowOSGroups** . Sin embargo, si se incluye la configuración del almacén de claves local, debe contener como mínimo:

- Vía de acceso de archivo **KeyStore** y
- Nombre de **CertLabel** .

```
AuthToken:
   KeyStore=/var/mqm/qmgrs/qmgrs/qm1/tokenissuer/key.kdb
   CertLabel=token
   ... up to 32 CertLabel fields
```

Si sólo ha incluido los dos parámetros mínimos, entonces:

- Debe existir un archivo de ocultación `key.sth` con la contraseña de almacén de claves cifrada para que el archivo de contraseña de almacén de claves no sea necesario.
- La señal no contiene un nombre de usuario que se debe pasar a IBM MQ para su autorización. La aplicación se puede conectar y autenticar, pero debe haber un mecanismo diferente para proporcionar autorización para que la aplicación funcione después de que se haya conectado.

En función de la configuración del gestor de colas, el nombre de usuario que se utiliza para la autorización puede ser el que se ha definido en el canal a través de reglas de MCA o el nombre de usuario que la aplicación cliente ha ejecutado tal como puede existir en el servidor y que pertenece a grupos con autorizaciones. Tenga en cuenta que cuando utilice señales:

- El gestor de colas se coloca en modalidad **UserExternal** , lo que significa que los usuarios que no existen en el sistema operativo donde se ejecuta el gestor de colas se pueden utilizar para la autenticación.
- Incluso si no incluye la opción **AllowOSGroups** en la stanza **AuthToken** `qm.ini` , el valor predeterminado se establece en No. Por lo tanto, si incluye **UserClaim** pero no especifica **AllowOSGroups**=Yes, el usuario de señal que se adopta para la autorización no se comprueba para los grupos a los que puede pertenecer en el sistema operativo donde se ejecuta el gestor de colas.

Stanza de ejemplo-autenticación y autorización

Puede definir todos los parámetros **AuthToken** :

- **KeyStore** vía de acceso de archivo,
- **KeyStorePwdFile** vía de acceso de archivo,
- **CertLabel** nombre,
- **UserClaim** y
- opción **AllowOSGroups**.

```
AuthToken:
   KeyStore=/var/mqm/qmgrs/qmgrs/QMJWT/ssl/key.kdb
   KeyStorePwdFile=/var/mqm/qmgrs/QMJWT/ssl/key.pw
   CertLabel=token
   CertLabel=rsakey
   CertLabel=mark
   ... up to 32 CertLabel fields
   UserClaim=AppUser
   AllowOSGroups=Y
```

Si ha incluido todos los parámetros disponibles, entonces:

- Cifre la contraseña para el almacén de claves utilizando el mandato **runqmcred** . Guárdelo en un archivo y, a continuación, incluya la vía de acceso del archivo en la stanza **AuthToken** .
- El nombre de usuario que está en la reclamación de usuario de señal de autenticación se utiliza para la autenticación y la autorización.
  - El usuario de señal puede existir como usuario en el sistema operativo donde se ejecuta el gestor de colas.
  - Ha definido un objeto de información de autenticación para habilitar la comprobación de usuario.
  - Los registros de autenticación de canal se configuran para adoptar un usuario con autorización para interactuar con objetos IBM MQ , basándose en las reglas de autenticación de canal o MCA.

La estrategia para autenticar y autorizar usuarios de señales depende de sus requisitos y de cómo ya estén configurados los gestores de colas de IBM MQ . Para obtener más información, consulte Trabajar con señales de autenticación.

Conceptos relacionados

Cómo trabajar con señales

Tareas relacionadas

Configuración de un gestor de colas para aceptar **AuthTokens**

Utilización de señales de autenticación en una aplicación

Multi Stanza AutoCluster del archivo qm.ini

La stanza AutoCluster se utiliza cuando el gestor de colas empieza a identificar si el clúster es miembro de un clúster automático y puede identificar los repositorios completos del clúster.

Los atributos siguientes son obligatorios para la stanza AutoCluster:

Type=*Uniform*
Especifica el tipo de clúster automático y la única opción válida es *Uniform*, que representa un clúster uniforme.

ClusterName=*<String>*
El nombre del clúster, que es el nombre de clúster automático.

Los atributos siguientes son opcionales para la stanza AutoCluster, pero debe proporcionarlos en pares:

RepositoryName1 =*< Serie>*
Se trata del nombre de gestor de colas para el primer repositorio completo del clúster automático. Este puede ser el nombre de este gestor de colas o de otro.

Repository1Conname=*< Serie de nombre de conexión >*
Se trata del valor de nombre de conexión (CONNAME) para la forma en que los miembros del clúster automático deben conectarse a este gestor de colas.

Repository2Name=*< Serie>*
Se trata del nombre de gestor de colas para el segundo repositorio completo del clúster automático. Este puede ser el nombre de este gestor de colas o de otro.

Repository2Conname=*< Serie de nombre de conexión >*
Se trata del valor de nombre de conexión (CONNAME) para la forma en que los miembros del clúster automático deben conectarse a este gestor de colas.

Stanza de ejemplo

```
AutoCluster:
        Repository1Name=QM1
        Repository2Name=QM2
        Repository1Conname=127.0.0.1(1414)
        Repository2Conname=127.0.0.1(1415)
        ClusterName=UNIFORMCLUSTER1
        Type=Uniform
```

Conceptos relacionados

“Equilibrio de aplicaciones automático” en la página 434
El equilibrado automático de aplicaciones mejora considerablemente la distribución y la disponibilidad de las aplicaciones habilitando un clúster uniforme de IBM MQ para gestionar de cerca la distribución de aplicaciones en todo el clúster, en lugar de depender de la aleatorización o de un anclaje manual de aplicaciones a gestores de colas específicos.

Tareas relacionadas

“Creación de un nuevo clúster uniforme” en la página 448
Cómo crear un nuevo clúster uniforme.

Referencia relacionada

“Utilización de la configuración de clúster automático” en la página 452
Puede configurar IBM MQ para habilitar la configuración automática cambiando la información de configuración de `qm.ini`.

Multi Stanza AutoConfig del archivo qm.ini

Los atributos de la stanza AutoConfig se suelen utilizar como parte de la configuración de clústeres uniformes.

Nota: Solo puede utilizar la stanza AutoCluster para clústeres uniformes.

MQSCConfig=*<Path>*
La vía de acceso es una vía de acceso de archivo completa o una vía de acceso a un directorio, donde todos los archivos `*.mqsc` se aplican al gestor de colas en cada inicio del gestor de colas.

Para obtener más información, consulte Configuración automática de un script de mandato de script de WebSphere MQ en el inicio.

IniConfig=*<Path>*
La vía de acceso es una vía de acceso de archivo completa o una vía de acceso a un directorio, donde todos los archivos `*.ini` se aplican al archivo `qm.ini` en cada inicio del gestor de colas.

Para obtener más información, consulte “Configuración automática de qm.ini en el inicio” en la página 110.

ConfigTimeout
Valor (en segundos) que el gestor de colas espera a que se complete la configuración automática. Transcurrido ese tiempo, el gestor de colas continúa el inicio y está disponible para que se conecten las aplicaciones.

El comportamiento predeterminada es no timeout. Esto significa que el gestor de colas no está disponible para que las aplicaciones se conecten hasta que se hayan completado todos los mandatos de configuración automática.

No debe configurar este atributo simplemente porque la configuración esté tardando mucho tiempo, ya que es posible que las aplicaciones se puedan conectar antes de que se haya completado la configuración aplicable a ellas, por ejemplo, la creación de las colas que requiere la aplicación.

Stanza de ejemplo

```
AutoConfig:
        MQSCConfig=/tmp/auto.mqsc
        IniConfig=/tmp/auto.ini
        ConfigTimeout=120
```

Conceptos relacionados

“Equilibrio de aplicaciones automático” en la página 434
El equilibrado automático de aplicaciones mejora considerablemente la distribución y la disponibilidad de las aplicaciones habilitando un clúster uniforme de IBM MQ para gestionar de cerca la distribución de aplicaciones en todo el clúster, en lugar de depender de la aleatorización o de un anclaje manual de aplicaciones a gestores de colas específicos.

Tareas relacionadas

“Creación de un nuevo clúster uniforme” en la página 448
Cómo crear un nuevo clúster uniforme.

Referencia relacionada

“Utilización de la configuración de clúster automático” en la página 452
Puede configurar IBM MQ para habilitar la configuración automática cambiando la información de configuración de `qm.ini`.

Multi Stanza de canales del archivo qm.ini

Los atributos de la stanza Channels determinan la configuración de un canal.

z/OS Esta información no es aplicable a IBM MQ for z/OS.

Utilice la stanza `CHANNELS` del archivo `qm.ini` para especificar información sobre canales.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades del gestor de colas de IBM MQ Explorer `Channels` .

MaxChannels = 100 (valor predeterminado) |*número*
El número máximo de canales *actuales* permitidos.

El valor por omisión es 100.

Puede establecer **MaxChannels** en un valor distinto para limitar el número máximo del canales actuales si es necesario. Para IBM MQ Appliance, el valor predeterminado es 999 999 999, y no debe cambiarse.

MaxActiveChannels=*valor_MaxChannels*
El número máximo de canales que pueden estar *activos* en cualquier momento. El valor predeterminado es el especificado en el atributo **MaxChannels**.

MaxInitiators = 3 (valor predeterminado) |*número*
El número máximo de iniciadores. El valor predeterminado y máximo es 3.

MQIBindType=FASTPATH|STANDARD
La vinculación para las aplicaciones:

FASTPATH
Los canales se conectan utilizando MQCONNX FASTPATH; es decir, no hay ningún proceso de agente.

ESTÁNDAR
Los canales se conectan utilizando STANDARD.

PipeLineLength=1|*número*
El número máximo de hebras simultáneas que utilizará un canal. El valor predeterminado es 1. Cualquier valor mayor que 1 se trata como 2.

Cuando utilice el proceso de canalización, debe configurar los gestores de colas en ambos extremos del canal para que tengan un valor de **PipeLineLength** mayor que 1.

Nota: El proceso de canalización solamente tiene efecto en los canales TCP/IP.

Consulte Soporte de varias hebras-interconexiones para obtener más información.

AdoptNewMCA = NO (valor predeterminado) | SVR | SDR | RCVR | CLUSRCVR | ALL | FASTPATH
Si IBM MQ recibe una solicitud para iniciar un canal, pero descubre que ya se está ejecutando una instancia del canal, en algunos casos la instancia del canal existente debe detenerse para poder iniciar la nueva instancia. El atributo **AdoptNewMCA** le permite controlar qué tipos de canales pueden finalizarse de esta manera.

Si especifica el atributo **AdoptNewMCA** para un tipo de canal concreto, pero el nuevo canal no se inicia correctamente debido a que una instancia de canal coincidente ya se está ejecutando:

1. El nuevo canal intenta detener el canal anterior con una solicitud para que finalice.
2. Si el servidor de canal anterior no responde a esta solicitud antes de que caduque el tiempo del intervalo de espera **AdoptNewMCATimeout**, finaliza la hebra o el proceso para el servidor de canal anterior.
3. Si el servidor de canal anterior no ha finalizado después del paso 2, y después de que caduque por segunda vez el intervalo de espera **AdoptNewMCATimeout**, IBM MQ finaliza el canal con un error del tipo CANAL EN USO.

La funcionalidad **AdoptNewMCA** se aplica a los canales de servidor, emisor, receptor y receptar de clúster. En el caso de un canal emisor o servidor, sólo puede ejecutarse una instancia de un canal con un nombre específico en el gestor de colas receptor. En el caso de un canal receptor o de clúster receptor, pueden ejecutarse varias instancias de un canal con un nombre específico en el gestor de colas receptor, pero en cualquier momento específico, sólo puede ejecutarse una instancia de un gestor de colas remoto específico.

Nota: AdoptNewMCA no está soportado en los canales peticionarios o de conexión de servidor.

Especifique uno o más valores, separados por comas o espacios en blanco, de la lista siguiente:

NO
La característica AdoptNewMCA no es necesaria. Este es el valor predeterminado.

SVR
Adoptar canales servidores.

SDR
Adoptar canales emisores.

RCVR
Adoptar canales receptores.

CLUSRCVR
Adoptar canales receptores de clúster.

TODOS
Adoptar todos los tipos de canales, excepto los canales FASTPATH.

FASTPATH
Adoptar el canal si se trata de un canal FASTPATH. Esto sólo ocurre si se especifica también el tipo de canal adecuado, por ejemplo: `AdoptNewMCA=RCVR,SVR,FASTPATH`.

¡Atención!: El atributo AdoptNewMCA puede comportarse de forma imprevisible con canales FASTPATH. Por lo tanto, tenga mucho cuidado al habilitar el atributo AdoptNewMCA para canales FASTPATH.

AdoptNewMCATimeout= 60 (valor predeterminado) | 1-3600
La cantidad de tiempo, en segundos, que la nueva instancia de canal deberá esperar a que finalice la instancia de canal anterior. Especifique un valor entre 1 y 3600. El valor predeterminado es 60.

AdoptNewMCACheck=QM|ADDRESS|NAME|ALL
El tipo de comprobación necesario cuando se habilita el atributo `AdoptNewMCA`. A ser posible, realice las comprobación completa para impedir que los canales se cierren accidental o intencionadamente. Como mínimo, compruebe que los nombres de los canales coinciden.

Especifique uno o más de los valores siguientes, separados por comas o espacios en blanco en el caso de *QM*, *NAME* o *ALL*:

QM
Compruebe que los nombres de los gestores de colas coinciden.

Tenga en cuenta que el nombre del gestor de colas en sí coincide, no el QMID.

ADDRESS
Compruebe la dirección de IP de origen de comunicaciones. Por ejemplo, la dirección TCP/IP.

Nota: Los valores CONNAME separados por coma se aplican a las direcciones de destino y, por consiguiente, no son relevantes para esta opción.

En el caso de que un gestor de colas multiinstancia falla desde `hosta` a `hostb`, los canales de salida de ese gestor de colas utilizará la dirección IP de origen `hostb`. Si esto es diferente de `hosta`, AdoptNewMCACheck=*ADDRESS* no coinciden.

Puede utilizar SSL o TLS con la autenticación mutua para evitar que un atacante interrumpa un canal en ejecución existente. Como alternativa, utilice una solución de tipo HACMP con toma de IP en lugar de gestores de colas multiinstancia o utilice un equilibrador de carga de red para enmascarar la dirección IP de origen.

NOMBRE
Compruebe que los nombres de los canales coinciden.

TODOS
Compruebe si coinciden los nombres de los gestores de colas, la dirección de comunicaciones y si coinciden los nombres de los canales.

El valor predeterminado es `AdoptNewMCACheck=NAME,ADDRESS,QM`.

ChlauthEarlyAdopt = Y (valor predeterminado) | N
El orden en que se procesan las reglas de autenticación de conexión y de canal es un factor importante a la hora de determinar el contexto de seguridad de las conexiones de aplicación de cliente de IBM MQ.

Atención: El valor predeterminado si **ChlauthEarlyAdopt** no está presente en el archivo qm.ini es N. Sin embargo, todos los gestores de colas se crean con **ChlauthEarlyAdopt**=Y añadido automáticamente al archivo qm.ini .

ChlauthEarlyAdopt solo adopta los ID de usuario que se han proporcionado a un gestor de colas para la autenticación de la conexión, si ADOPTCTX(YES) está establecido en el objeto AUTHINFO de autenticación de conexión en el gestor de colas.

Los valores válidos de **ChlauthEarlyAdopt** son los siguientes:

Y
El canal valida y adopta las credenciales de ID de usuario y contraseña proporcionadas por una aplicación que usa la autenticación de conexión del gestor de colas antes de aplicar las reglas de autenticación de canal. En este modo de operación, las reglas de autenticación de canal se emparejan con el ID de usuario resultante de las comprobaciones de autenticación de conexión.

N
El canal aplaza la validación de autenticación de conexión de las credenciales de ID de usuario y contraseña proporcionadas por una aplicación hasta después de que se hayan aplicado las reglas de autenticación de canal. Tenga en cuenta que este modo de operación, el bloqueo de la autenticación de canal y las reglas de correlación no puede tener en cuenta el resultado de la validación del ID de usuario y la contraseña.

Por ejemplo, el objeto de información de autenticación predeterminado se establece en **ADOPTCTX(YES)** y el usuario `fred` inicia una sesión. Se configuran estas dos reglas CHLAUTH:

```
SET CHLAUTH('MY.CHLAUTH') TYPE(ADDRESSMAP) DESCR('Block all access by
default') ADDRESS('*') USERSRC(NOACCESS) ACTION(REPLACE)
SET CHLAUTH('MY.CHLAUTH') TYPE(USERMAP) DESCR('Allow user bob and force
CONNAUTH') CLNTUSER('bob') CHCKCLNT(REQUIRED) USERSRC(CHANNEL)
```

Se emite el siguiente mandato, con la intención de autenticar el mandato como el contexto de seguridad adoptado del usuario bob:

```
runmqsc -c -u bob QMGR
```

De hecho, el gestor de colas utiliza el contexto de seguridad `fred`, no bob, y la conexión falla.

Para utilizar el contexto de seguridad de bob, **ChlauthEarlyAdopt** debe establecerse en Y.

PasswordProtection = Compatible (valor predeterminado) |always|opcional|warn

Las credenciales de autenticación que las aplicaciones de IBM MQ client especifican cuando se conectan a un gestor de colas se pueden proteger utilizando la característica de protección de contraseña MQCSP de IBM MQ , si la conexión no utiliza el cifrado TLS.

La protección por contraseña MQCSP es útil para fines de prueba y desarrollo porque utilizar la protección por contraseña MQCSP es más sencillo que establecer el cifrado TLS, pero no es tan seguro.

Para obtener más información sobre la protección de credenciales en la estructura MQCSP y los valores que se pueden establecer para este atributo, consulte Protección de contraseña MQCSP.

IgnoreSeqNumberMismatch = NO (valor predeterminado) | YES

Los agentes de canal de mensajes en los dos extremos de un canal llevan cada uno el recuento del número de mensajes enviados a través del canal para mantener la sincronización. La sincronización se puede perder, por ejemplo, si la definición de canal en un extremo se suprime y luego se vuelve a crear. En estas circunstancias, es posible que se necesite un RESET CHANNEL para reconocer que los datos de sincronización se han perdido y permitir que el canal continúe el inicio.

El atributo **IgnoreSeqNumberMismatch** se debe establecer en el gestor de colas receptor.

En la práctica, este atributo realiza un mandato de restablecimiento de canal en el canal receptor.

Este atributo controla la forma en que el gestor de colas maneja la no coincidencia de número de secuencia durante el inicio del canal utilizando los valores siguientes:

NO
Los números de secuencia de canal se comprueban durante la resincronización de canal, si los dos MCA no coinciden en el mismo número de secuencia, se notificará el mensaje de error AMQ9526 y no se podrá iniciar el canal.

SÍ
Los números de secuencia de canal se comprueban durante la resincronización de canal, pero si los dos MCA no coinciden en el mismo número de secuencia, se notificará el mensaje de aviso AMQ9703 y continuará el inicio del canal. Este valor de atributo no debería ser necesario en circunstancias normales. Cuando se sabe que se han perdido datos de sincronización, por ejemplo, durante la recuperación tras desastre, esta opción evita la necesidad de reconocer manualmente cada no coincidencia de número de secuencia. La especificación de este valor tiene un efecto similar al de un administrador que emite automáticamente un **RESET CHANNEL** en respuesta a cada discrepancia de número de secuencia.

ChlauthIgnoreUserCase = N (valor predeterminado) | Y
Permite que un gestor de colas haga la coincidencia del nombre de usuario dentro de las reglas CHLAUTH sin distinción entre mayúsculas y minúsculas. Esta opción permite:

- CLNTUSER en reglas CHLAUTH TYPE(USERMAP) que coincida sin distinción entre mayúsculas y minúsculas
- USERLIST en reglas CHLAUTH TYPE(BLOCKUSER) que coincida sin distinción entre mayúsculas y minúsculas

Los valores válidos de **ChlauthIgnoreUserCase** son los siguientes:

N
Las reglas de autenticación de canal intentan hacer coincidir la identificación de usuario cliente con distinción entre mayúsculas y minúsculas, por ejemplo una regla que especifica CLNTUSER ('Fred') no coincidirá con 'fred' o 'FRED', sólo coincidirá con un identificador de usuario de 'Fred'. Éste es el valor predeterminado.

Y
Las reglas de autenticación de canal intentan hacer coincidir la identificación de usuario cliente con la no distinción entre mayúsculas y minúsculas, por ejemplo, una regla de autenticación de canal con TYPE(USERMAP) o TYPE(USERBLOCK) que especifica CLNTUSER('Fred') coincidirá con cualquier variación de mayúsculas y minúsculas, por ejemplo los identificadores de usuario 'Fred', 'FRED' y 'fred' todos coinciden.

Tenga en cuenta que, al ignorar las mayúsculas y minúsculas de los identificadores de usuario cuando coinciden las reglas de autenticación de canal, es posible que más de una regla coincida. Si esto ocurre, la regla que se compara está sin definir. Por ejemplo, con las reglas siguientes, si el usuario 'fred' se conecta a un gestor de colas a través del canal CLIENT, se pueden correlacionar con 'mquser1' o 'mquser2':

```
SET CHLAUTH('CLIENT') TYPE(USERMAP) CLNTUSER('fred') USERSRC(MAP) MCAUSER('mquser1')
SET CHLAUTH('CLIENT') TYPE(USERMAP) CLNTUSER('FRED') USERSRC(MAP) MCAUSER('mquser2')
```

Para evitar cualquier incertidumbre al utilizar `ChlauthIgnoreUserCase=Y`, evite definir reglas CHLAUTH que se solapen y den como resultado un comportamiento diferente al utilizar una coincidencia sin distinguir entre mayúsculas y minúsculas.

ChlauthIssueWarn = y
Establezca este atributo si desea que se genere el mensaje AMQ9787 al establecer el atributo WARN = YES en el mandato **SET CHLAUTH** .

Stanza de ejemplo

```
Channels:
   MaxChannels=200
   MaxActiveChannels=100
   MQIBindType=STANDARD
```

```
    PipelineLength=2
```

Conceptos relacionados

“Estados de un canal” en la página 242

Un canal puede estar en cualquier momento en uno de los muchos estados que existen. Algunos estados también tienen subestados. A partir de un estado determinado un canal puede pasar a otros estados.

Multi Stanza Connection del archivo qm.ini

La stanza Connection define el tipo de enlace predeterminado.

Utilice la stanza Connection del archivo `qm.ini` para especificar el tipo de enlace predeterminado.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades del gestor de colas de IBM MQ Explorer `Extended` .

Nota: Debe crear una stanza Connection si necesita una.

DefaultBindTipo = SHARED (valor predeterminado) |AISLADO

Si **DefaultBindType** está establecido en ISOLATED, las aplicaciones y el gestor de colas se ejecutan en procesos separados, y no se comparte ningún recurso entre ellos.

Si **DefaultBindType** está establecido en SHARED, las aplicaciones y el gestor de colas se ejecutan en procesos separados, pero algunos recursos se comparten entre ellos.

El valor predeterminado es SHARED.

Atención: DefaultBindType se aplica a todas las llamadas MQCONN y a cualquiera que utilice MQCONNX con MQCNO_STANDARD_BINDING.

El cambio de **DefaultBindType** puede hacer que se degrade el rendimiento de algunas aplicaciones.

Stanza de ejemplo

```
Connection:
  DefaultBindType=SHARED
```

Multi Registro de mensajes de diagnóstico

Los registros de mensajes de diagnóstico de IBM MQ son un mecanismo para permitir que varios componentes del sistema IBM MQ notifiquen mensajes de diagnóstico relacionados con la configuración de IBM MQ y problemas y cambios de estado de tiempo de ejecución.

Estos registros a veces se conocen como IBM MQ *registros de errores*, pero siempre han contenido IBM MQ información y mensajes de aviso, así como mensajes de error. Los tres componentes principales de IBM MQ que informan en estos registros son:

- Gestores de colas
- Clientes de IBM MQ
- El resto del sistema IBM MQ

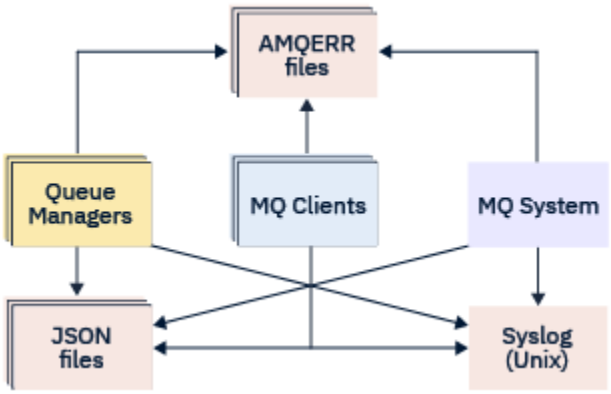
IBM MQ admite la creación de informes de mensajes de diagnóstico a través de una serie de distintos métodos conocidos como *servicios de mensajes de diagnósticos*, que permiten un enfoque adaptado para registrar y consumir esta información.

- Archivos de registro AMQERRnn
- Archivos de registro con formato JSON
- AIX Syslog en formato JSON

La salida JSON de IBM MQ se formatea como objetos JSON de una sola línea como, de forma que cada línea individual del registro JSON, o el registro Syslog, representa un objeto JSON válido. El registro como un todo no se encapsula como un único objeto JSON.

La ilustración siguiente muestra que los gestores de colas, los clientes de IBM MQ y el sistema IBM MQ pueden *todos* notificar mensajes de diagnóstico utilizando los métodos descritos.

Figura 5. Cómo pueden notificar mensajes de diagnóstico distintas parte de IBM MQ



Cómo se configuran los registros de diagnósticos de IBM MQ:

Los registros de diagnóstico se definen y personalizan utilizando stanzas dentro del archivo `qm.ini` en particular para el componente IBM MQ que los necesita. Cada punto final de registro exclusivo se define bajo su propia cabecera de stanza dentro del archivo ini, junto con cualquier personalización definida en el mismo. Las personalizaciones pueden incluir:

- El tamaño de los archivos de registro para recortar, antes de que produzca el aplazamiento; no es aplicable al Syslog
- Cualquier filtrado basado en la gravedad de los mensajes de registro, y
- Cualquier código de mensaje específico para suprimir.

IBM MQ se puede configurar para grabar en cualquiera, o en todos, de los tres tipos de puntos finales de registro, lo que permite que stanzas de registro concretas puedan cumplir roles determinados. De forma similar, se pueden definir varios servicios de archivos. Por ejemplo:

- El formato JSON facilita el análisis a través de un conjunto de herramientas automatizadas en entornos locales y de nube.
- La salida del Syslog permite a los componentes IBM MQ integrar la información de diagnóstico en una ubicación de registro de SO común de acuerdo con otros productos en el sistema.
- Puntos finales de registro basados en la gravedad que permiten que se registren archivos de registro determinados, por ejemplo, solo errores graves en el sistema.

Independientemente del estilo de registro de diagnóstico configurado, siempre se escriben los archivos de diagnóstico tradicionales que se mantienen bajo el directorio de registro del sistema IBM MQ (`/var/mqm/errors/AMQERRnn.log`) y el directorio de registro del gestor de colas específico (`/var/mqm/qmgrs/<qmgr_name>/errors/AMQERRnn.log`), además de cualquier otra configuración de registro utilizada.

Solo para gestores de colas, la configuración opcional de estos registros obligatorios se puede realizar especificando atributos de “Stanzas del servicio de mensajes de diagnóstico” en la página 137.

Áreas de stanza diferentes

Las stanzas adicionales se pueden aplicar en distintas áreas de IBM MQ.

Gestor de colas(qm.ini)
Se aplica al mensaje de registro generado por el gestor de colas.

Sistema(mqs.ini)
Se aplica al os mensajes de registro generados por el sistema. Esta opción no es específica a un gestor de colas, excepto cuando un gestor de colas no puede acceder o grabar en sus propios registros.

Plantillas(mqs.ini)
Una o más stanzas como plantillas, que se copian en `qm.ini` cuando se crea un gestor de colas.

Cliente(mqclient.ini)
Se aplica a la operación de cliente, por ejemplo **`runmqsc`** en la modalidad de cliente para un gestor de colas remoto.

Conversión entre registros con formato JSON y con formato tradicional

El mandato mqrc se ha mejorado para permitir una serie de conversiones entre registros con formato JSON y con formato tradicional, y entre distintos lenguajes.

Referencia relacionada

"Stanzas del servicio de mensajes de diagnóstico" en la página 137
Las opciones del servicio de mensaje de diagnóstico disponibles permiten la personalización del registro de diagnósticos de IBM MQ, de forma que la salida del registro se puede direccionar a distintos puntos finales de registro desde diferentes componentes de IBM MQ.

"Stanza QMerrorLog" en la página 136
Utilice la stanza de registro de errores del gestor de colas QMerrorLog en el archivo `qm.ini` para adaptar la operación y el contenido de los registros de errores de IBM MQ.

"Servicios de mensajes de diagnóstico" en la página 140
Los servicios de mensaje de diagnóstico siguientes y sus atributos específicos de servicio, especificados bajo las stanzas DiagnosticSystemMessages, DiagnosticMessages y DiagnosticMessagesTemplate de los archivos de configuración se pueden definir:

Multi *Stanza QMerrorLog*

Utilice la stanza de registro de errores del gestor de colas QMerrorLog en el archivo `qm.ini` para adaptar la operación y el contenido de los registros de errores de IBM MQ.

El servicio QMErrorLog es el servicio de registro de diagnóstico de IBM MQ tradicional utilizado para generar mensajes de diagnósticos que pertenecen al gestor de colas. El servicio QMErrorLog se ejecuta de forma continuada y no se puede desactivar, pero se puede personalizar hasta cierto punto.

Puede utilizar la stanza QMerrorLog del archivo `qm.ini` para excluir determinados mensajes de ser escritos en el registro de errores del gestor de colas. También puede suprimir mensajes de ser escritos en el registro de errores para un intervalo de tiempo determinado.

Windows Linux De forma alternativa, en lugar de editar el archivo `qm.ini` directamente, puede utilizar la página de propiedades de Extended Queue Manager en IBM MQ Explorer para excluir y suprimir mensajes con los atributos **Excluded messages**, **Suppressed messages** y **Suppressed messages interval**.

Atención:

- Windows Puede utilizar IBM MQ Explorer para realizar los cambios solo si está utilizando un gestor de colas local en la plataforma Windows.
- La stanza QMerrorLog no es aplicable al archivo de configuración del sistema IBM MQ, `mqs.ini` o al archivo de configuración del cliente, generalmente denominado `mqclient.ini`.

Los atributos siguientes se pueden incluir en la stanza QMErrorLog:

ErrorLogSize=*tamañoMáx*
Especifica el tamaño del registro de errores del gestor de colas que se copia en la copia de seguridad. *tamañoMáx* debe estar comprendido entre el rango de 32768 a 2147483648 bytes. Si no se especifica **ErrorLogSize**, se utiliza el valor predeterminado de 33554432 bytes (32 MB).

Este atributo se puede usar para reducir el tamaño máximo al anterior máximo de 2 MB, si es necesario.

Puede establecer el tamaño del registro utilizando la variable de entorno **MQMAXERRORLOGSIZE** .

ExcludeMessage= *msgIds*
Especifica mensajes que no se deben grabar en el registro de errores del gestor de colas.

Consulte ExcludeMessage en "Stanzas del servicio de mensajes de diagnóstico" en la página 137 si desea más información.

SuppressMessage= *msgIds*
Especifica que se graben mensajes en el registro de errores del gestor de colas sólo una vez en un intervalo de tiempo especificado. Si se especifica el mismo ID de mensaje en `SuppressMessage` y `ExcludeMessage`, el mensaje se excluye.

Esta opción no es aplicable a los servicios de mensajes de diagnóstico definidos en `mqclient.ini`. Para obtener más información, consulte SuppressMessage en "Stanzas del servicio de mensajes de diagnóstico" en la página 137.

SuppressInterval= *longitud*
Especifica el intervalo de tiempo, en segundos, en el que los mensajes especificados en `SuppressMessage` se graban en el registro de errores del gestor de colas sólo una vez. *longitud* debe ser un valor comprendido entre 1 y 86400 segundos. Si no se especifica `SuppressInterval`, se utiliza el valor predeterminado de 30 segundos.

Stanza de ejemplo

```
QMErrorLog:
   ErrorLogSize=262144
   ExcludeMessage=7234
   SuppressMessage=9001,9002,9202
   SuppressInterval=30
```

Conceptos relacionados

"Archivos de configuración de gestores de colas, qm.ini" en la página 109
Un archivo de configuración del gestor de colas, `qm.ini`, contiene información relevante para un gestor de colas específico. Atributos que se pueden utilizar para modificar la configuración de un gestor de colas individual y sustituir lo valores de IBM MQ.

Referencia relacionada

"Stanzas del servicio de mensajes de diagnóstico" en la página 137
Las opciones del servicio de mensaje de diagnóstico disponibles permiten la personalización del registro de diagnósticos de IBM MQ, de forma que la salida del registro se puede direccionar a distintos puntos finales de registro desde diferentes componentes de IBM MQ.

Multi *Stanzas del servicio de mensajes de diagnóstico*

Las opciones del servicio de mensaje de diagnóstico disponibles permiten la personalización del registro de diagnósticos de IBM MQ, de forma que la salida del registro se puede direccionar a distintos puntos finales de registro desde diferentes componentes de IBM MQ.

Habilite servicios de mensaje de diagnóstico adicionales, utilizando una stanza con uno de los nombres siguientes:

- **DiagnosticSystemMessages**

  Define los servicios utilizados cuando se genera un mensaje de diagnóstico dirigido al registro de errores del sistema. Válido en los archivos `mqs.ini` o `mqclient.ini`.

Las aplicaciones cliente utilizan la stanza **DiagnosticSystemMessages** en los archivos mqclient.ini y mqs.ini, la stanza **DiagnosticSystemMessages** controla los mensajes de una aplicación de servidor que no tiene un contexto de gestor de colas.

Es posible configurar un gestor de colas y aplicaciones que también escriban todos los mensajes en el servicio syslog.

- **DiagnosticMessages**

  Define los servicios utilizados cuando se genera un mensaje de diagnóstico dirigido al registro de errores del gestor de colas. Sólo es válido en el archivo `qm.ini`.

- **DiagnosticMessagesTemplate**

  Stanza que se copia del archivo `mqs.ini` a **DiagnosticMessages** en el archivo `qm.ini` cuando se crea un gestor de colas.

Para visualizar los mensajes de diagnóstico, utilice el mandato mqrc.

Atributos de las stanzas

Atención: `Service` y el nombre de una stanza son obligatorios.

name=<nombrestanza>
Nombre de una stanza. El valor debe ser exclusivo en un archivo ini.

Service= *tipo de servicio*

Este atributo define un servicio, donde el nombre del servicio no distingue entre mayúsculas y minúsculas, que está habilitando esta stanza.

Por ejemplo, para habilitar `syslog` como un servicio adicional, especifique lo siguiente:

```
Service=syslog
```

Consulte “Servicios de mensajes de diagnóstico” en la página 140 y sus atributos específicos que están disponibles para ser utilizados con las stanzas del servicio de mensaje de diagnóstico.

Se pueden añadir los siguientes atributos opcionales a las stanzas:

- ExcludeMessage
- SuppressMessage
- SuppressInterval
- “Severities” en la página 140

ExcludeMessage= *msgIds*
Especifica mensajes que no se deben grabar en el registro de errores del gestor de colas. Si el sistema IBM MQ se utiliza de forma intensiva, con muchos canales que se detienen e inician, se envía un gran número de mensajes informativos al registro de copia impresa y a la consola de z/OS. El puente IBM MQ - IMS y el gestor de almacenamiento intermedio también pueden producir un gran número de mensajes informativos, por lo que excluir mensajes le impide recibir un gran número de mensajes si lo necesita. *msgIds* contiene una lista separada por comas de los ID de mensaje de lo siguiente:

5211 - Se ha sobrepasado la longitud máxima del nombre de propiedad.
5973 - Suscripción de publicación/suscripción distribuida inhibida
5974 - Publicación de publicación/suscripción distribuida inhibida
6254 - El sistema no ha podido cargar la biblioteca compartida de forma dinámica
IBM i 7163 - Mensaje de trabajo iniciado (sólo IBM i)
7234 - Número de mensajes cargados
8245 - La entidad no tiene autoridad suficiente para mostrar el objeto
9001 - El programa del canal ha finalizado normalmente
9002 - El canal del programa se ha iniciado

9202 - Host remoto no disponible
9208 - Error al recibir del host
9209 - Conexión cerrada
9228 - No se puede iniciar el programa de respuesta de canal
9489 - Se ha sobrepasado el límite máximo de instancias SVRCONN
9490 - Se ha sobrepasado el límite máximo de instancias SVRCONN por cliente
9508 - No se puede conectar al gestor de colas
9524 - Gestor de colas remoto no disponible
9528 - El usuario ha solicitado el cierre del canal
9545 - Ha expirado el intervalo de desconexión
9558 - El canal remoto no está disponible
9637 - Al canal le falta un certificado
9776 - El ID de usuario bloqueó el canal
9777 - La correlación NOACCESS ha bloqueado el canal
9782 - La dirección ha bloqueado la conexión
9999 - El programa de canal ha finalizado de forma anómala

SuppressMessage= *msgIds*

Especifica que se graben mensajes en el registro de errores del gestor de colas sólo una vez en un intervalo de tiempo especificado. Si el sistema IBM MQ se utiliza de forma intensiva, con muchos canales que se detienen e inician, se envía un gran número de mensajes informativos al registro de copia impresa y a la consola de z/OS. El puente IBM MQ - IMS y el gestor de almacenamiento intermedio también pueden producir un gran número de mensajes informativos, por lo que suprimir mensajes le impide recibir una serie de mensajes repetitivos si lo necesita. El intervalo de tiempo se especifica con `SuppressInterval`. *msgIds* contiene una lista separada por comas de los identificadores de mensaje de lo siguiente:

5211 - Se ha sobrepasado la longitud máxima del nombre de propiedad.
5973 - Suscripción de publicación/suscripción distribuida inhibida
5974 - Publicación de publicación/suscripción distribuida inhibida
6254 - El sistema no ha podido cargar la biblioteca compartida de forma dinámica
IBM i 7163 - Mensaje de trabajo iniciado (sólo IBM i)
7234 - Número de mensajes cargados
8245 - La entidad no tiene autoridad suficiente para mostrar el objeto
9001 - El programa del canal ha finalizado normalmente
9002 - El canal del programa se ha iniciado
9202 - Host remoto no disponible
9208 - Error al recibir del host
9209 - Conexión cerrada
9228 - No se puede iniciar el programa de respuesta de canal
9489 - Se ha sobrepasado el límite máximo de instancias SVRCONN
9490 - Se ha sobrepasado el límite máximo de instancias SVRCONN por cliente
9508 - No se puede conectar al gestor de colas
9524 - Gestor de colas remoto no disponible
9528 - El usuario ha solicitado el cierre del canal
9545 - Ha expirado el intervalo de desconexión
9558 - El canal remoto no está disponible
9637 - Al canal le falta un certificado
9776 - El ID de usuario bloqueó el canal
9777 - La correlación NOACCESS ha bloqueado el canal
9782 - La dirección ha bloqueado la conexión
9999 - El programa de canal ha finalizado de forma anómala

Si se especifica el mismo ID de mensaje en `SuppressMessage` y `ExcludeMessage`, el mensaje se excluye.

Esta opción no es aplicable a los servicios de mensajes de diagnóstico definidos en `MQ client.ini`.

SuppressInterval= *longitud*
Especifica el intervalo de tiempo, en segundos, en el que los mensajes especificados en **SuppressMessage** se graban en el registro de errores del gestor de colas sólo una vez. *longitud* tiene que encontrarse en el rango de 1 - 86400 segundos. Si no se especifica **SuppressInterval**, se utiliza el valor predeterminado de 30 segundos.

Severities
Lista separada por comas de los niveles de gravedad, en cuyos nombres de nivel de gravedad no se distingue entre mayúsculas y minúsculas. Los valores permitidos son:

- I (o Information o 0)
- W (o Warning o 10)
- E (o Error o 20 y 30)
- S (o Stop o 40)
- T (o System o 50)

Notas:

1. El valor predeterminado es `all` (todos)
2. Solo los mensajes de los niveles de gravedad seleccionados se presentan al servicio.

   De forma alternativa, se puede usar el carácter más (+) que representa el nivel de error especificado y todos los niveles superiores. Por ejemplo, para visualizar todos los errores:

```
Severities=E+
```

Referencia relacionada
“Stanza QMerrorLog” en la página 136
Utilice la stanza de registro de errores del gestor de colas QMerrorLog en el archivo `qm.ini` para adaptar la operación y el contenido de los registros de errores de IBM MQ.

“Servicios de mensajes de diagnóstico” en la página 140
Los servicios de mensaje de diagnóstico siguientes y sus atributos específicos de servicio, especificados bajo las stanzas DiagnosticSystemMessages, DiagnosticMessages y DiagnosticMessagesTemplate de los archivos de configuración se pueden definir:

Multi *Servicios de mensajes de diagnóstico*
Los servicios de mensaje de diagnóstico siguientes y sus atributos específicos de servicio, especificados bajo las stanzas DiagnosticSystemMessages, DiagnosticMessages y DiagnosticMessagesTemplate de los archivos de configuración se pueden definir:

Se han definido los siguientes servicios de mensajes de diagnóstico:

Archivo
Este servicio envía los mensajes no filtrados a un archivo de forma similar al servicio QMErrorLog. Se utiliza el formato textual existente o el formato JSON especificado según el parámetro **Format** especificado. De forma predeterminada, hay tres archivos denominados `AMQERR01.LOG`, `AMQERR02.LOG`y `AMQERR03.LOG` o `AMQERR01.json`, `AMQERR02.json`y `AMQERR03.json`, en función de la propiedad **Format** y de estos aplazamientos en función del tamaño configurado.

Los siguientes atributos solo están soportados en una stanza File:

FilePath
La vía de acceso donde se van a escribir los archivos de registro. El valor predeterminado es la misma ubicación que los archivos `AMQERR01.log`, es decir, el sistema o el gestor de colas. La vía de acceso debe ser absoluta, pero puede incluir inserciones sustituibles. Por ejemplo:

+MQ_Q_MGR_DATA_PATH+
La vía de acceso completa del padre del directorio de mensajes de diagnósticos del gestor de colas. Los valores predeterminados son:

- Linux AIX En plataformas AIX and Linux: `/var/mqm/qmgrs/<QM_name>`
- Windows en Windows, `C:\Program Data\IBM\MQ\qmgrs\<QM_name>`

+MQ_DATA_PATH+
La vía de acceso completa del padre del directorio de mensajes de diagnósticos del sistema. Los valores predeterminados son:

- Linux AIX En plataformas AIX and Linux:`/var/mqm`
- Windows En Windows: `C:\Program Data\IBM\MQ`

Debe crear esta vía de acceso con los permisos apropiados, si no está utilizando el directorio de errores existente.

FilePrefix
Prefijo de los archivos de registro. El valor predeterminado es AMQERR.

FileSize
El tamaño con el que se rota el registro. El valor predeterminado es 32 MB, como con la propiedad **ErrorLogSize** de "Stanza QMerrorLog" en la página 136, que es semánticamente idéntico.

Nota: La propiedad **ErrorLogSize** sólo se aplica al servicio de registro de errores predeterminado, no a los servicios de diagnóstico personalizados.

Puede establecer el tamaño del registro utilizando la variable de entorno **MQMAXERRORLOGSIZE** .

Format
El formato del archivo. El valor puede ser *text* (para servicios de estilo QMErrorLog adicionales) o *json*, que es el valor predeterminado.

El sufijo del archivo es `.LOG` o `.json`, según el valor de este atributo.

Por ejemplo, edite el archivo `qm.ini` del gestor de colas y añada la siguiente stanza:

```
DiagnosticMessages:
   Service = File
   Name = JSONLogs
   Format = json
   FilePrefix = AMQERR
```

Después de reiniciar, el gestor de colas tendrá archivos `AMQERR0x.json` en su directorio ERRORS.

Se puede definir varios servicios de archivos. Esto permite la configuración que se muestra en los ejemplos siguientes, donde mensajes de distintas etiquetas se reparten entre distintos conjuntos de registros:

```
DiagnosticMessages:
  Name=ErrorsToFile
  Service=File
  Severities=E+
  FilePrefix=OnlyErrors

DiagnosticMessages:
  Name=NonErrorstoFile
  Service=File
  Severities=1 W
  FilePrefix=Information
```

Linux AIX **Syslog**
El servicio Syslog no está disponible en Windows o IBM i

Solo puede definir un servicio Syslog, y el servicio Syslog envía los mensajes sin filtrar al syslog utilizando la especificación de mensajes de diagnóstico de formato JSON. La información se añade al syslog en el orden que se muestra en la tabla, empezando por el msgID y las inserciones.

La gravedad del mensaje se correlaciona con el nivel syslog de la siguiente manera:

Tabla 13. Correlación de la gravedad de mensaje con el nivel de syslog

Gravedad	**Nivel**
0	LOG_INFO
10	LOG_WARNING
20	LOG_ERR
30	LOG_ERR
40	LOG_ALERT
50	LOG_ALERT

El atributo siguiente solo está soportado en una instancia syslog:

Ident
Define el valor de **ident** asociado a las entradas de syslog. El valor predeterminado es *ibm-mq*.

El ejemplo siguiente muestra mensajes de error que se están enviando al Syslog:

```
DiagnosticMessages:
  Name=ErrorsToSyslog
  Ident=mq
  Service=Syslog
  Severities=E+
```

Consulte "Stanzas del servicio de mensajes de diagnóstico" en la página 137 si desea más información sobre atributos de stanza genéricos.

Notas:

1. Solo en el servicio de archivo, se pueden tener varias stanzas, cada una con un nombre distinto. Solo entra en vigor la última definición de la secuencia.
2. Los cambios en el valor de una stanza solo entran en vigor cuando se reinicia el gestor de colas.

Multi Stanza ExitPath del archivo qm.ini

La stanza ExitPath especifica la vía de acceso para los programas de salida de usuario en el sistema del gestor de colas.

Utilice la stanza `ExitPath` del archivo `qm.ini` para especificar la vía de acceso para los programas de salida de usuario en el sistema del gestor de colas.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades del gestor de colas de IBM MQ Explorer `Exits` .

ExitsDefaultPath=*serie*
El atributo ExitsDefaultPath especifica la ubicación de:

- Salidas de canal de 32 bits para clientes
- Salidas de canal de 32 bits y salidas de conversión de datos para servidores
- Archivos de carga conmutada XA no calificados

ExitsDefaultPath64=*serie*
El atributo ExitsDefaultPath64 especifica la ubicación de:

- Salidas de canal de 64 bits para clientes
- Salidas de canal de 64 bits y salidas de conversión de datos para servidores
- Archivos de carga conmutada XA no calificados

Stanza de ejemplo

```
ExitPath:
   ExitsDefaultPath=/var/mqm/exits
   ExitsDefaultPath64=/var/mqm/exits64
```

Multi Stanza ExitPropertiesLocal del archivo qm.ini

La stanza local ExitPropertiesespecifica información sobre las propiedades de salida en un gestor de colas.

Utilice la stanza ExitPropertieslocal en el archivo `qm.ini` , para especificar información sobre las propiedades de salida en un gestor de colas.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades del gestor de colas de clúster IBM MQ Explorer .

Windows De forma alternativa, en Windows puede especificar esta información utilizando el mandato **amqmdain** .

De forma predeterminada, este valor se hereda del atributo **CLWLMode** en la stanza ExitProperties de la configuración de toda la máquina (que se describe en “Stanza ExitProperties del archivo mqs.ini” en la página 104). Cambie este valor sólo si desea configurar el gestor de colas de forma diferente. Este valor se puede alterar temporalmente para gestores de colas individuales utilizando el atributo de modalidad de carga de trabajo del clúster en la página de propiedades de gestor de colas Clúster.

Utilice la stanza ExitProperties del archivo `mqs.ini` para especificar las opciones de configuración utilizadas por los programas de salida del gestor de colas.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades de IBM MQ Explorer `Extended` IBM MQ .

CLWLMode = SAFE (valor predeterminado) | FAST
La salida de carga de trabajo de clúster (CLWL) le permite especificar qué cola de clúster del clúster se debe abrir en respuesta a una llamada MQI (por ejemplo, `MQOPEN`, `MQPUT`). La salida CLWL se ejecuta en modalidad FAST o en modalidad SAFE en función del valor que especifique en el atributo **CLWLMode** . Si omite el atributo **CLWLMode** , la salida de carga de trabajo del clúster se ejecuta en modalidad SAFE.

SAFE
Ejecutar la salida CLWL en un proceso distinto al del gestor de colas. Este es el valor predeterminado.

Si surge algún problema con la salida CLWL escrita por el usuario mientras se está ejecutando en modalidad SAFE, se producirá lo siguiente:

- El proceso del servidor CLWL (amqzlwa0) no se ejecutará correctamente.
- El gestor de colas reiniciará el proceso del servidor CLWL.
- El error se indicará en los registros de error. Si hay una llamada MQI en proceso, se recibirá una notificación en forma de código de retorno.

Se mantiene la integridad del gestor de colas.

Nota: La ejecución de la salida CLWL en un proceso independiente puede afectar al rendimiento.

FAST
Ejecutar la salida de clúster incorporada en el proceso del gestor de colas.

Especificar esta opción mejora el rendimiento al evitar los costes de conmutación de proceso que implica la ejecución en modalidad SAFE, pero esto se produce a expensas de la integridad del gestor de colas. Sólo debe ejecutar la salida CLWL en modalidad FAST si está convencido de que no hay problemas con la salida CLWL y está especialmente preocupado por el rendimiento.

Si surge algún problema cuando la salida CLWL está ejecutándose en modalidad FAST, el gestor de colas no se ejecutará correctamente y correrá el riesgo de comprometer la integridad del gestor de colas.

Stanza de ejemplo

```
ExitPropertiesLocal:
      CLWLMode=SAFE
```

IBM i Linux AIX Stanza de sistema de archivos del archivo qm.ini

La stanza Filesystem especifica si los permisos establecidos en los registros de errores del gestor de colas deben permanecer sin cambios o deben volver a cambiarse a sus valores predeterminados.

Se espera que los permisos predeterminados establecidos en los archivos de registro de errores sean útiles en la mayoría de las circunstancias y, por lo tanto, no es necesario que la mayoría de los administradores de IBM MQ los alteren.

Sin embargo, es posible que el administrador de IBM MQ desee modificar los permisos en sus archivos de registro de errores, en cuyo caso deben establecer la opción de stanza `Filesystem` **ValidateAuth**=*No*, lo que hace que el gestor de colas deje los permisos sin modificar posteriormente.

El comportamiento predeterminado (sin **ValidateAuth**=*No*) es que el gestor de colas comprueba los permisos de archivo de los registros de errores del gestor de colas y los cambia de nuevo a sus valores predeterminados. Esta comprobación puede suceder en cualquier momento, incluso durante una operación de inicio o finalización del gestor de colas.

Stanza de ejemplo

```
Filesystem:
      ValidateAuth=No
```

Linux V 9.4.0 AIX Stanza JWKS del archivo mq.ini

Utilice la stanza **JWKS** para informar al gestor de colas sobre cómo recuperar un conjunto de claves públicas que pueden validar señales firmadas por este emisor.

EndPoint=*serie*
Especifica el URL de un documento JWKS que proporciona las claves públicas necesarias para validar las señales firmadas por este emisor.

Este servidor especificado debe proporcionar una conexión segura (HTTPS) para recuperar las claves. El gestor de colas también debe haberse configurado para establecer conexiones HTTP seguras, consulte HTTPSKeyStore.

Si no está seguro del valor correcto para este atributo, consulte al administrador del servidor de autenticación.

IssuerName=*serie*
Debe ser el **IssuerName** que está presente en las señales de autenticación JWT proporcionadas/firmadas por este emisor.

Puede establecer esto revisando la documentación del servidor de autenticación o examinando un JWT de ejemplo emitido por el servicio.

UserClaim=*serie*
Esto proporciona el nombre de una reclamación (par de clave/valor dentro de una señal de autenticación) que el gestor de colas utilizará como ID de usuario de IBM MQ al establecer las autorizaciones de IBM MQ .

Si las señales de este punto final se utilizan sólo para la autenticación, es decir, **ADOPTCTX** es NO, este atributo es opcional; de lo contrario, es necesario.

Stanza de ejemplo

```
JWKS:
    Endpoint=https://myauthserver.example/jwks
    IssuerName=https://myauthserver.example/jwks
    UserClaim=MQUser
```

Información relacionada
Configuración de un gestor de colas para aceptar señales de autenticación utilizando un punto final JWKS

Multi Stanza de registro del archivo qm.ini

La stanza Log especifica información sobre el registro en un gestor de colas.

Utilice la stanza Log del archivo `qm.ini` para especificar información sobre el registro en un gestor de colas.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades IBM MQ Explorer **Gestor de colas de registro** .

De forma predeterminada, estos valores se heredan de los valores especificados para los valores de registro predeterminados para el gestor de colas (descritos en “Stanza LogDefaults del archivo mqs.ini” en la página 105). Solamente debe modificar estos valores si desea configurar este gestor de colas de un modo distinto.

Para obtener más información sobre el cálculo de tamaños de registro, consulte “Cálculo del tamaño del registro” en la página 680.

Nota: Los límites que se proporcionan en la siguiente lista de parámetros los establece IBM MQ. Los límites del sistema operativo podrían reducir el tamaño de registro máximo posible.

LogPrimaryFiles = 3 (valor predeterminado) |*2-254* (Windows) |*2-510* (sistemas AIX and Linux)
Los archivos de anotaciones asignados cuando se crea el gestor de colas.

El número mínimo de archivos de registro primarios que puede tener es 2 y el máximo es 254 en Windows, o 510 en sistemas AIX and Linux. El valor predeterminado es 3.

El número total de archivos de registro primarios y secundarios no debe superar 255 en Windows, o 511 en sistemas AIX and Linux y no debe ser inferior a 3.

Cuando se crea o inicia el gestor de colas, se examina el valor. Puede cambiarlo después de haber creado el gestor de colas. No obstante, si modifica el valor, el cambio no entra en vigor hasta que se reinicia el gestor de colas, y es posible que el efecto no sea inmediato.

LogSecondaryFiles = 2 (valor predeterminado) |*1-253* (Windows) |*1-509* (sistemas AIX and Linux)
Los archivos de anotaciones que se asignan cuando se agotan los archivos primarios.

El número mínimo de archivos de registro secundarios es 1 y el máximo es 253 en Windows, o 509 en sistemas AIX and Linux. El valor predeterminado es 2.

El número total de archivos de registro primarios y secundarios no debe superar 255 en Windows, o 511 en sistemas AIX and Linux y no debe ser inferior a 3.

El valor se examina cuando se inicia el gestor de colas. Puede modificar este valor, pero los cambios no surtirán efecto hasta que se reinicie el gestor de colas, y es posible que el efecto no sea inmediato.

LogFilePages=*número*
Los datos de las anotaciones se guardan en una serie de archivos llamados archivos de anotaciones. El tamaño del archivo de registro se especifica en unidades de páginas de 4 KB.

El número predeterminado de páginas de archivo de registro es 4096, lo que da un tamaño de archivo de registro de 16 MB.

En sistemas AIX and Linux, el número mínimo de páginas de archivo de registro es 64, y en Windows, el número mínimo de páginas de archivo de registro es 32; en ambos casos, el número máximo es 65535.

Nota: El tamaño de los archivos de registro especificados durante la creación del gestor de colas no se puede cambiar para un gestor de colas.

LogType = CIRCULAR (valor predeterminado) | LINEAR| REPLICATED
El tipo de registro que utilizará el gestor de colas. El valor predeterminado es `CIRCULAR`. Para obtener más información sobre cómo crear un gestor de colas con el tipo de registro que necesita, consulte la descripción del atributo **LogType** en "Stanza LogDefaults del archivo mqs.ini" en la página 105.

CIRCULAR
Inicie la recuperación de reinicio utilizando el registro para retrotraer las transacciones que estaban en curso cuando se detuvo el sistema.

Consulte "Tipos de registro" en la página 674 para ver una explicación completa del registro circular.

LINEAR
Este valor permite efectuar tanto la recuperación de reinicio como la recuperación desde soporte o por repetición de actualizaciones (creando los datos perdidos o dañados mediante la reproducción del contenido del registro).

En el apartado "Tipos de registro" en la página 674 puede ver una explicación completa sobre las anotaciones cronológicas lineales.

REPLICATED
Lo utiliza el grupo de HA nativa para replicar datos de registro de la instancia activa a las instancias de réplica.

Consulte "Tipos de registro" en la página 674 para ver una explicación más completa del registro replicado.

Nota: El **LogType** de un gestor de colas no se puede cambiar modificando este atributo en el archivo `qm.ini`. Para cambiar el **LogType** de un gestor de colas, debe utilizar el mandato **migmqlog**.

LogBufferPages=0 (valor predeterminado) |*0-4096*
La cantidad de memoria asignada a los registros de almacenamiento intermedio para grabación, especificando el tamaño de los almacenamientos intermedios en unidades de páginas de 4 KB.

El número mínimo de páginas de almacenamiento intermedio es de 18 y el número máximo es de 4.096. Los almacenamientos intermedios más grandes dan como resultado un rendimiento superior, especialmente para mensajes grandes.

Si especifica 0 (el valor predeterminado), el gestor de colas selecciona el tamaño.

Si especifica un número entre 1 y 17, el gestor de colas toma de forma predeterminada el valor de 18 (72 KB). Si especifica un número en el rango de 18 a 4096, el gestor de colas utiliza el número especificado para establecer la cantidad de memoria asignada.

El valor se examina cuando se inicia el gestor de colas. El valor se puede aumentar o disminuir dentro de los límites estipulados. Sin embargo, el cambio del valor no entra en vigor hasta que se inicia de nuevo el gestor de colas.

LogPath=*nombre_directorio*
El directorio en el que residen los archivos de registro de un gestor de colas. Este directorio debe existir en un dispositivo local en el que el gestor de colas pueda grabar y, preferiblemente, en una unidad que no sea la que contiene las colas de mensajes. Especificando una unidad distinta se consigue una protección adicional por si se produce una anomalía en el sistema.

El valor predeterminado es:

- Windows `C:\ProgramData \IBM \MQ\log` en Windows.
- Linux AIX `/var/mqm/log` en sistemas AIX and Linux.

Puede especificar el nombre de un directorio en el mandato **crtmqm** utilizando el distintivo **-ld** . Al crear un gestor de colas, también se crea un directorio debajo del directorio del gestor de colas, que se utiliza para contener los archivos de registros. El nombre de este directorio se basa en el nombre del gestor de colas. Esto asegura que la vía de acceso del archivo de registro sea exclusiva y que cumpla con los límites establecidos para la longitud de nombres de directorios.

Si no especifica `-ld` en el mandato **crtmqm** , se utiliza el valor del atributo **LogDefaultPath** .

Linux AIX En sistemas AIX and Linux , el ID de usuario mqm y el grupo mqm deben tener autorizaciones completas sobre los archivos de registro. Si cambia las ubicaciones de estos archivos, debe otorgar usted mismo estas autorizaciones. Esto no es necesario si los archivos de registro están en las ubicaciones predeterminadas suministradas con el producto.

LogWriteIntegrity =SingleWrite|DoubleWrite|TripleWrite (valor predeterminado)
El método que utiliza el registrador de anotaciones para grabar los registros de anotaciones de forma fiable.

TripleWrite (valor predeterminado)

Observe que puede seleccionar `DoubleWrite`, pero si lo hace, el sistema los interpreta como `TripleWrite`.

SingleWrite
Debe utilizar `SingleWrite`, solo si el sistema de archivos y el dispositivo que aloja el registro de recuperación de IBM MQ garantiza de forma explícita la atomicidad de escrituras de 4 KB.

Es decir, cuando una escritura de una página de 4KB falla por algún motivo, los dos únicos estados posibles son la imagen anterior o la imagen posterior. No debería ser posible ningún estado intermedio.

Nota: Si hay suficiente simultaneidad en la carga de trabajo persistentes, hay un mínimo beneficio potencial en establecer cualquier otra cosa que el valor predeterminado, `TripleWrite`.

Para obtener más información, consulte “LogWriteIntegridad-utilizando SingleWrite o TripleWrite” en la página 148.

LogManagement = Manual (predeterminado) | Automático | Archivado
El método utilizado para gestionar las extensiones de registro, ya sea manualmente o mediante el gestor de colas. El valor predeterminado es `Manual`.

El atributo sólo se aplica cuando **LogType** es `LINEAR`.

Si cambia el valor de **LogManagement**, el cambio no entrará en vigor hasta que se reinicie el gestor de colas.

Si se encuentra un valor no reconocido para el atributo, el gestor de colas no se iniciará hasta que se corrija el valor.

IBM i La propiedad **LogManagement** no es válida en IBM i.

manual (valor predeterminado)
Las extensiones de registro se gestionan manualmente. Si se especifica esta opción significa que el gestor de colas no reutilizará ni suprimirá las extensiones de registro, incluso si ya no son necesarias para la recuperación.

Automático
El gestor de colas gestiona automáticamente las extensiones de registro. Si se especifica esta opción significa que el gestor de colas puede reutilizar o suprimir las extensiones de registro en cuanto ya no sean necesarias para la recuperación. No se realiza ninguna asignación para el archivado.

Archivo
El gestor de colas gestiona las extensiones de registro, pero debe avisar al gestor de colas cuando haya finalizado el archivado de cada extensión de registro.

Especificar esta opción significa que el gestor de colas puede reutilizar o suprimir una extensión de registro, siempre que se haya notificado al gestor de colas que se ha archivado una extensión que ya no es necesaria para la recuperación.

Esta notificación se realiza utilizando el mandato MQSC de **RESET QMGR** o el mandato PCF Restablecer gestor de colas .

Stanza de ejemplo

```
Log:
   LogPrimaryFiles=3
   LogSecondaryFiles=2
   LogFilePages=4096
   LogType=CIRCULAR
   LogBufferPages=0
   LogPath=/var/mqm/log/saturn!queue!manager/
```

Nota: El valor de cero para **LogBufferPages** le otorga un valor de 512.

Multi ***LogWriteIntegridad-utilizando SingleWrite o TripleWrite***

El establecimiento de la opción **LogWriteIntegrity**, en la stanza Log del archivo `qm.ini`, determina el algoritmo que utiliza el registrador en IBM MQ para grabar los registros en el registro de recuperación. El valor predeterminado es `TripleWrite` y este valor es seguro en casi todos los escenarios posibles.

El valor de **LogWriteIntegrity** tiene cualquier efecto solo, cuando se va a escribir una página de registro parcial. Para un gestor de colas con una cantidad razonable de actividad simultánea, este escenario se produce raramente.

SingleWrite

`SingleWrite` selecciona un algoritmo que, bajo circunstancias muy inusuales, puede ejecutarse mejor que el valor `TripleWrite` predeterminado. El valor de `SingleWrite` es seguro, solo si la plataforma de almacenamiento subyacente puede garantizar de forma absoluta en todas las circunstancias que las páginas de 4KB escritas de forma síncrona en el registro de recuperación de IBM MQ se escriben de forma atómica.

Debe utilizar el valor `SingleWrite` , sólo si el sistema de archivos o dispositivo, que aloja el registro de recuperación de IBM MQ , garantiza explícitamente la atomicidad de las grabaciones de 4KB . Es decir, cuando falla una grabación de una página de 4 KB por cualquier motivo, los dos únicos estados posibles deben ser la imagen anterior o la imagen posterior, y no debe ser posible ningún estado intermedio. En todos los demás casos, debe utilizar `TripleWrite`.

En un sistema con simultaneidad suficiente, el gestor de colas solo graba páginas completas de datos de registro, y si se consigue un alto porcentaje de páginas completas, no hay ninguna diferencia de rendimiento significativa entre `SingleWrite` y `TripleWrite`.

En un sistema con poca simultaneidad, puede haber una ventana de rendimiento significativa en `SingleWrite`, sin embargo, la solución preferida suele ser aumentar la simultaneidad, en lugar de utilizar `SingleWrite`.

Tenga en cuenta que puede ser difícil determinar de forma fiable la atomicidad de las grabaciones de 4 KB y los cambios en el software o hardware subyacente podrían invalidar dicha garantía.

Si tiene alguna duda de que la infraestructura de almacenamiento de almacenamiento realiza las garantías necesarias ahora y, en cualquier momento en el futuro bajo todas las circunstancias, debe utilizar `TripleWrite`.

Windows Stanza LU62 del archivo qm.ini (solo Windows)

La stanza LU62 especifica los parámetros de configuración del protocolo SNA LU 6.2 . Estos parámetros alteran temporalmente los atributos predeterminados para los canales.

Utilice la stanza LU62 del archivo `qm.ini` para especificar los parámetros de configuración del protocolo SNA LU 6.2 . Alteran temporalmente los atributos predeterminados de los canales.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades del gestor de colas de IBM MQ Explorer LU6.2 .

Nombre TP
El nombre de TP que debe iniciarse en la ubicación remota.

Library1=*NombreDLL1*
El nombre de la DLL de APPC.

El valor predeterminado es WCPIC32.

Library2= *DLLName2*
Igual que Library1, utilizada si el código se almacena en dos bibliotecas distintas.

El valor predeterminado es WCPIC32.

CP4I Stanza `NativeHAInstance` del archivo qm.ini

Para IBM MQ en contenedores, la stanza `NativeHAInstance` especifica cómo los tres nodos de una configuración de HA nativa pueden comunicarse entre sí.

Nota: Esta información sólo se aplica a entornos de contenedor. Consulte Configuración de HA nativa utilizando el operador IBM MQ o Creación del grupo HA nativa si crea sus propios contenedores.

Añada tres stanzas `NativeHAInstance` , una para cada instancia de gestor de colas en el grupo HA nativo (incluida la instancia local). Añada los atributos siguientes:

Nombre
Especifique el nombre de instancia que ha utilizado al crear la instancia del gestor de colas.

ReplicationAddress
Especifique el nombre de host, IPv4 decimal con puntos o IPv6 dirección de formato hexadecimal de la instancia. Puede especificar la dirección como un nombre de host, IPv4 decimal con puntos o dirección en formato hexadecimal IPv6 . La dirección de réplica debe poder resolverse y direccionarse desde cada instancia del grupo. El número de puerto que se debe utilizar para la réplica de registro debe especificarse entre corchetes, por ejemplo:

```
ReplicationAddress=host1.example.com(4444)
```

Stanza de ejemplo

El ejemplo siguiente muestra la stanza `NativeHAInstance` utilizada en el archivo `qm.ini` para especificar los tres nodos de una configuración de HA nativa.

```
NativeHAInstance:
  Name=node-1
```

```
   ReplicationAddress=host1.example.com(4444)
 NativeHAInstance:
   Name=node-2
   ReplicationAddress=host2.example.com(4444)
 NativeHAInstance:
   Name=node-3
   ReplicationAddress=host3.example.com(4444)
```

Conceptos relacionados

“Stanza NativeHALocalInstance del archivo qm.ini” en la página 150
Para IBM MQ en contenedores, la stanza `NativeHALocalInstance` controla el funcionamiento de una configuración de HA nativa.

CP4I Stanza `NativeHALocalInstance` del archivo qm.ini

Para IBM MQ en contenedores, la stanza `NativeHALocalInstance` controla el funcionamiento de una configuración de HA nativa.

Nota: Esta información sólo se aplica a entornos de contenedor. Consulte Configuración de HA nativa utilizando el operador IBM MQ o Creación del grupo HA nativa si crea sus propios contenedores.

La stanza `NativeHALocalInstance` se añade automáticamente al archivo `qm.ini` en cada uno de los nodos al crear una configuración de HA nativa. A continuación, puede editar el archivo `qm.ini` y personalizar los atributos en la stanza `NativeHALocalInstance` .

`LocalName`
El nombre de la stanza `NativeHALocalInstance` , tomado del nombre de instancia de réplica de registro especificado cuando se crea el gestor de colas de HA nativa.

Opcionalmente, puede añadir los atributos siguientes a la stanza `NativeHALocalInstance` :

`KeyRepository`
La vía de acceso completa y el nombre de archivo del repositorio de claves que contiene el certificado digital que se utiliza para proteger el tráfico de réplica de registro. Si no se especifica la extensión de archivo, se presupone que es `.kdb`.

Si se omite el atributo de stanza `KeyRepository` , los datos de réplica de registro se intercambian entre instancias en texto sin formato.

V 9.4.0 `KeyRepositoryPassword`

El repositorio de claves está protegido con una contraseña, ya que contiene información confidencial. Para poder acceder al contenido del repositorio de claves, IBM MQ debe poder recuperar la contraseña del repositorio de claves. Si la contraseña no se almacena en un archivo de ocultación de repositorio de claves, puede proporcionar la contraseña en el atributo `KeyRepositoryPassword` . Por ejemplo:

```
KeyRepositoryPassword=passw0rd
```

Atención: Si proporciona la contraseña utilizando este atributo, cifre la contraseña con el sistema de protección de contraseñas de IBM MQ . Para obtener más información, consulte “Cifrado de la contraseña del repositorio de claves” en la página 152.

V 9.4.0 `InitialKeyFile`

Especifique este atributo si la contraseña del repositorio de claves que se especifica con el atributo `KeyRepositoryPassword` se cifra con una clave inicial específica. El nombre del archivo que contiene la clave inicial se puede especificar utilizando el parámetro **`-sf`** cuando se utiliza el mandato **`runmqicred`** para cifrar la contraseña del repositorio de claves.

Establezca el valor de este atributo en el nombre del archivo que contiene la clave inicial utilizada para cifrar la contraseña. Por ejemplo, si un archivo denominado `mykey.key` contiene la clave inicial:

```
InitialKeyFile=/mykey.key
```

Para obtener más información, consulte “Cifrado de la contraseña del repositorio de claves” en la página 152.

CertificateLabel
Etiqueta de certificado que identifica el certificado digital que se debe utilizar para la protección del tráfico de réplica de registro. Si se proporciona KeyRepository pero se omite CertificateLabel , se utiliza un valor predeterminado de ibmwebspheremq*queue_manager* .

CipherSpec
La CipherSpec que se debe utilizar para proteger el tráfico de réplica de registro. Si se proporciona este atributo de stanza, también se debe proporcionar KeyRepository . Si se proporciona KeyRepository pero se omite CipherSpec , se utiliza un valor predeterminado de ANY .

LocalAddress
La dirección de la interfaz de red local que acepta el tráfico de réplica de registro. Si se proporciona este atributo de stanza, identifica la interfaz de red local y/o el puerto utilizando el formato "[addr][(port)]". La dirección de red se puede especificar como un nombre de host, IPv4 decimal con puntos o formato hexadecimal IPv6 . Si se omite este atributo, el gestor de colas intenta enlazar con todas las interfaces de red, utiliza el puerto especificado en ReplicationAddress en la stanza NativeHAInstances que coincide con el nombre de instancia local.

HeartbeatInterval
El intervalo de latidos define la frecuencia en milisegundos a la que una instancia activa de un gestor de colas de HA nativo envía una pulsación de red. El rango válido del valor del intervalo de pulsaciones es de 500 (0.5 segundos) a 60000 (1 minuto), un valor fuera de este rango hace que el gestor de colas no se pueda iniciar. Si se omite este atributo, se utiliza un valor predeterminado de 5000 (5 segundos). Cada instancia debe utilizar el mismo intervalo de pulsaciones.

HeartbeatTimeout
El tiempo de espera de latido define cuánto tiempo espera una instancia de réplica de un gestor de colas de HA nativo antes de decidir que la instancia activa no responde. El rango válido del valor de tiempo de espera de intervalo de pulsaciones es de 500 (0.5 segundos) a 120000 (2 minutos). El valor del tiempo de espera de pulsaciones debe ser mayor o igual que el intervalo de pulsaciones.

Un valor no válido hace que el gestor de colas no se inicie. Si se omite este atributo, una réplica espera 2 x HeartbeatInterval antes de iniciar el proceso para seleccionar una nueva instancia activa. Cada instancia debe utilizar el mismo tiempo de espera de latido.

RetryInterval

El intervalo de reintento define la frecuencia en milisegundos a la que un gestor de colas HA nativo debe reintentar un enlace de réplica anómalo. El rango válido del intervalo de reintento es de 500 (0.5 segundos) a 120000 (2 minutos). Si se omite este atributo, una réplica espera 2 x HeartbeatInterval antes de reintentar un enlace de réplica fallido.

SSLFipsRequired

Especifica si sólo se utilizan algoritmos certificados por FIPS si se utiliza la criptografía en el envío del tráfico de réplica de registro. Establézcalo en Yes o No.

EncryptionPolicySuiteB

Especifica si el tráfico de réplica de registro utiliza criptografía compatible con Suite-B y qué nivel de intensidad se utiliza. Establézcalo en uno de los valores siguientes:

NONE
No se utiliza el cifrado compatible con Suite B. Este valor es el predeterminado.

128_BIT,192_BIT
Establece el nivel de seguridad para niveles de 128 bit y de 192 bits.

128_BIT
Establece la potencia de seguridad en un nivel de 128 bits.

192_BIT
Establece la potencia de seguridad en un nivel de 192 bits.

V 9.4.0 **CompressionThreshold**
Establece un umbral de bytes que, cuando se cruza, desencadena la compresión de los datos de registro. Los datos de registro mayores que el valor de umbral se comprimen. Un valor de 0 (el valor predeterminado asumido) desactiva toda la compresión, un valor de 1 comprime cada anotación cronológica añadida. El valor máximo es 268435456 (256 MB).

V 9.4.0 **LZ4Acceleration**
Parámetro de ajuste que controla cómo el algoritmo LZ4 busca secuencias comprimibles en los datos de registro. Cada vez que el valor de aceleración aumenta en 1, el algoritmo no se ve tan cuidadosamente para una secuencia comprimible, intercambiando la relación de compresión por una pequeña ganancia de rendimiento. El valor mínimo (y el valor predeterminado asumido) es 1, el máximo es 65537.

Cifrado de la contraseña del repositorio de claves

V 9.4.0

La contraseña del repositorio de claves se puede proteger utilizando el sistema de protección de contraseñas de IBM MQ o un archivo de ocultación de repositorio de claves. Para obtener más información sobre estos dos métodos, consulte Cifrado de contraseñas de repositorio de claves.

Si la contraseña del repositorio se especifica utilizando el atributo `KeyRepositoryPassword` en la stanza `NativeHALocalInstance` , cifre la contraseña utilizando el sistema de protección de contraseñas IBM MQ . Utilice el mandato **runmqicred** para cifrar la contraseña. El mandato devuelve la contraseña cifrada que se puede especificar en el atributo `KeyRepositoryPassword` .

Utilice una clave inicial exclusiva para cifrar la contraseña de forma segura. El nombre del archivo que contiene la clave inicial se puede especificar utilizando el parámetro **-sf** en el mandato **runmqicred** . Si no proporciona una clave exclusiva, se utiliza la clave predeterminada.

Si cifra la contraseña del repositorio de claves con una clave inicial exclusiva, también debe proporcionar la misma clave inicial utilizando el atributo `InitialKeyFile` en la stanza `NativeHALocalInstance` .

Stanza de ejemplo

El ejemplo siguiente muestra la stanza `NativeHALocalInstance` utilizada en el archivo `qm.ini` para especificar el nombre local de un nodo.

```
NativeHALocalInstance:
  LocalName=node-1
```

Conceptos relacionados

"Stanza NativeHAInstance del archivo qm.ini" en la página 149
Para IBM MQ en contenedores, la stanza `NativeHAInstance` especifica cómo los tres nodos de una configuración de HA nativa pueden comunicarse entre sí.

Referencia relacionada

runmqicred (proteger contraseñas de cliente IBM MQ)

Windows Stanza NETBIOS del archivo qm.ini (solo Windows)

La stanza NETBIOS del archivo `qm.ini` especifica los parámetros de configuración del protocolo NetBIOS . Estos parámetros alteran temporalmente los atributos predeterminados para los canales.

Utilice la stanza NETBIOS del archivo `qm.ini` para especificar los parámetros de configuración del protocolo NetBIOS . Alteran temporalmente los atributos predeterminados de los canales.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades del gestor de colas de Netbios de IBM MQ Explorer .

LocalName= ***nombre***
El nombre por el que se conoce a la máquina en la LAN.

AdapterNum = 0 (valor predeterminado) | *número_adaptador*
El número del adaptador de la LAN. El valor predeterminado es el adaptador 0.

NumSess = 1 (valor predeterminado) | *número_de_sesiones*
El número de sesiones que se debe asignar. El valor predeterminado es 1.

NumCmds = 1 (valor predeterminado) | *número_de_mandatos*
El número de mandatos que se debe asignar. El valor predeterminado es 1.

NumNames = 1 (valor predeterminado) | *número_de_nombres*
El número de nombres que se debe asignar. El valor predeterminado es 1.

Library1= *DLLName1*
El nombre de la DLL de NetBIOS.

El valor predeterminado es NETAPI32.

Conceptos relacionados

"Definición del nombre NETBIOS local de IBM MQ" en la página 278
El nombre NetBIOS local que los procesos de canal de IBM MQ utilizan se puede especificar de tres modos.

Linux AIX Stanza RestrictedMode del archivo qm.ini

La stanza RestrictedMode especifica el nombre del grupo que contiene miembros que pueden ejecutar aplicaciones MQI, actualizar todos los recursos IPCC y cambiar el contenido de algunos directorios del gestor de colas. Esta stanza sólo se aplica a sistemas AIX and Linux .

La stanza RestrictedMode se establece mediante la opción **-g** en el mandato **crtmqm**. Si no utiliza la opción **-g**, la stanza no se creará en el archivo `qm.ini`.

Hay algunos directorios bajo los cuales las aplicaciones de IBM MQ crean archivos mientras están conectadas al gestor de colas dentro del directorio de datos del gestor de colas. Para que las aplicaciones puedan crear archivos en estos directorios, se les otorga acceso de grabación global:

- `/var/mqm/sockets/`*`QMgrName`*`/@ipcc/ssem/`*`hostname`*`/`
- `/var/mqm/sockets/`*`QMgrName`*`/@app/ssem/`*`hostname`*`/`
- `/var/mqm/sockets/`*`QMgrName`*`/zsocketapp/`*`hostname`*`/`

donde *QMGRNAME* es el nombre del gestor de colas y *hostname* es el nombre de host.

En algunos sistemas, es inaceptable conceder a todos los usuarios acceso de grabación para dichos directorios. Por ejemplo, aquellos usuarios que no necesiten acceder al gestor de colas. La modalidad restringida modifica los permisos de los directorios en los que se almacenan los datos del gestor de colas. Por tanto, sólo pueden acceder a los directorios los miembros del grupo de aplicaciones especificado. Los permisos de la memoria compartida de System V IPC utilizada para comunicarse con el gestor de colas, también se modifican del mismo modo.

El grupo de aplicaciones es el nombre del grupo con miembros que tienen permiso para realizar las acciones siguientes:

- Ejecutar aplicaciones MQI
- Actualizar todos los recursos IPCC
- Cambiar el contenido de algunos directorios de gestores de colas

Para utilizar la modalidad restringida para un gestor de colas:

- El creador del gestor de colas debe estar en el grupo `mqm` y en el grupo de aplicaciones.
- El ID de usuario `mqm` debe estar en el grupo de aplicaciones.
- Todos los usuarios que deseen administrar el gestor de colas deben estar en el grupo `mqm` y en el grupo de aplicaciones.
- Todos los usuarios que deseen ejecutar aplicaciones IBM MQ deben estar en el grupo de aplicaciones.

Cualquier llamada MQCONN o MQCONNX que emita un usuario que no esté en el grupo de aplicaciones falla, con el código de razón MQRC_Q_MGR_NOT_AVAILABLE.

Importante: En muchos sistemas operativos, para que se reconozca la adición de un usuario a un grupo, el usuario en cuestión debe finalizar la sesión y volver a iniciarla.

La modalidad restringida funciona con el servicio de autorización de IBM MQ. Por tanto, también debe otorgar a los usuarios la autorización para conectarse a IBM MQ y acceder a los recursos que necesiten utilizar el servicio de autorización de IBM MQ.

ALW Puede encontrar más información sobre cómo configurar el servicio de autorización de IBM MQ en Configuración de la seguridad en sistemas AIX, Linux, and Windows.

Sólo utilice la modalidad restringida de IBM MQ cuando el control proporcionado por el servicio de autorización no proporcione un aislamiento suficiente de los recursos del gestor de colas.

Referencia relacionada

crtmqm (crear gestor de colas)

Multi Stanza Security del archivo qm.ini

La stanza Security especifica opciones para el Gestor de autorizaciones sobre objetos (OAM).

ClusterQueueAccessControl=RQMName|Xmitq

Establezca este atributo para comprobar el control de acceso de las colas de clúster o las colas totalmente calificadas alojadas en los gestores de colas de clústeres.

RQMName

Los perfiles cuyo control de accesos de colas alojadas de forma remota se comprueba son colas con nombre o perfiles del gestor de colas con nombre.

Xmitq

Los perfiles cuyo control de accesos de colas alojadas de forma remota se comprueba se resuelven en `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

`Xmitq` es el valor predeterminado.

Windows **GroupModel=GlobalGroups**

Este atributo determina si el OAM comprueba grupos globales cuando se determina la pertenencia a grupos de un usuario en Windows.

El valor predeterminado es no comprobar los grupos globales.

GlobalGroups

El OAM comprueba los grupos globales.

Con `GlobalGroups` establecido, los mandatos de autorización, **setmqaut**, **dspmqaut**y **dmpmqaut** aceptan nombres de grupos globales; consulte el parámetro **setmqaut** -g .

Nota: El establecimiento de `ClusterQueueAcessControl=RQMName` y tener una implementación predeterminada del servicio de autorización en menos de `MQZAS_VERSION_6` da como resultado que el gestor de colas no se inicie. En este ejemplo, establezca `ClusterQueueAcessControl=Xmitq` o aumente el servicio de autorización personalizado a `MQZAS_VERSION_6` o superior.

Stanza de ejemplo

```
Security:
      ClusterQueueAccessControl=Xmitq
      GroupModel=GlobalGroups
```

Multi Stanza de servicio del archivo qm.ini

La stanza Service se utiliza para realizar cambios en los servicios instalables. Esta stanza contiene el nombre del servicio y el número de puntos de entrada definidos para el servicio.

Nota: Windows Linux La modificación de los servicios instalables y sus componentes tiene implicaciones importantes. Por este motivo, los servicios instalables son de sólo lectura en IBM MQ Explorer.

Para cada componente de un servicio, debe especificar también el nombre y la vía de acceso del módulo que contiene el código de dicho componente. Utilice la stanza ServiceComponent para ello.

Las stanzas **Service** y **ServiceComponent** pueden aparecer en cualquier orden y las claves de stanza bajo las mismas pueden aparecer en cualquier orden. Para cualquiera de estas stanzas, deben estar presentes todas las claves de stanza. Si se duplica una clave de stanza, se utilizará la última.

Durante el inicio, el gestor de colas procesa en turnos las entradas del componente de servicio que hay en el archivo de configuración. A continuación, carga el módulo de componente especificado, invocando el punto de entrada del componente, que debe ser el punto de entrada para la inicialización del componente, y lo pasa al manejador de configuración.

Name = AuthorizationService (valor predeterminado) |NameService
El nombre del servicio necesario.

AuthorizationService
Para IBM MQ, el componente **AuthorizationService** se conoce como gestor de autorizaciones sobre objetos u OAM. La stanza **Service** y su stanza **ServiceComponent** asociada se añaden automáticamente cuando se crea el gestor de colas, pero se pueden alterar temporalmente mediante la variable de entorno *MQSNOAUT* . Las demás stanzas **ServiceComponent** deben añadirse manualmente.

Linux AIX Los ejemplos siguientes de stanzas en el archivo `qm.ini` definen dos componentes de servicio de autorización en IBM MQ for AIX. *MQ_INSTALLATION_PATH* representa el directorio de alto nivel en el que está instalado IBM MQ.

```
Service:
   Name=AuthorizationService
   EntryPoints=13

ServiceComponent:
   Service=AuthorizationService
   Name=MQSeries.UNIX.auth.service
Module=MQ_INSTALLATION_PATH/lib/amqzfu
   ComponentDataSize=0

ServiceComponent:
   Service=AuthorizationService
   Name=user.defined.authorization.service
   Module=/usr/bin/udas01
   ComponentDataSize=96
```

Linux AIX La stanza ServiceComponent `MQSeries.UNIX.auth.service` define el componente de servicio de autorización predeterminado, el OAM. Si elimina esta stanza y reinicia el gestor de colas, el OAM se inhabilita y no se realiza ninguna comprobación de autorización.

Windows También puede añadir el atributo **SecurityPolicy** utilizando los servicios de IBM MQ. El atributo **SecurityPolicy** sólo se aplica si el servicio especificado en la stanza Service es el servicio de autorización, es decir, el OAM predeterminado. El atributo **SecurityPolicy** permite especificar la política de seguridad para cada gestor de colas. Los valores posibles son:

Valor predeterminado
Especifique `Default` si desea que la política de seguridad predeterminada entre en vigor. Si no se pasa un identificador de seguridad de Windows (SID de NT) al OAM para un ID de

usuario determinado, se intenta obtener el SID adecuado buscando en las bases de datos de seguridad pertinentes.

NTSIDsRequired
Requiere que se pase SID de NT al OAM al efectuar las comprobaciones de seguridad.

Windows La stanza ServiceComponent `MQSeries.WindowsNT.auth.service` define el componente de servicio de autorización predeterminado, el OAM. Si elimina esta stanza y reinicia el gestor de colas, el OAM se inhabilita y no se realiza ninguna comprobación de autorización.

NameService
No se proporciona ningún servicio de nombres de forma predeterminada. Si necesita un servicio de nombres, debe añadir manualmente la stanza NameService .

Linux AIX Los ejemplos siguientes de stanzas de archivo de AIX and Linux `qm.ini` para el servicio de nombres especifican un componente de servicio de nombres proporcionado por la compañía ABC (ficticia).

```
# Stanza for name service
 Service:
    Name=NameService
    EntryPoints=5

# Stanza for name service component, provided by ABC
 ServiceComponent:
    Service=NameService
    Name=ABC.Name.Service
    Module=/usr/lib/abcname
    ComponentDataSize=1024
```

Nota: Windows En sistemas Windows , la información de stanza NameService se almacena en el registro.

EntryPoints=*número-de-entradas*
El número de puntos de entrada definidos para el servicio.

Esto incluye los puntos de entrada de inicialización y terminación.

Windows **SecurityPolicy= Default|NTSIDsRequired**
En sistemas Windows , el atributo **SecurityPolicy** sólo se aplica si el servicio especificado es el servicio de autorización predeterminado, es decir, el OAM. El atributo **SecurityPolicy** permite especificar la política de seguridad para cada gestor de colas.

Los valores posibles son:

Valor predeterminado
Es el valor que se utiliza para que la política de seguridad predeterminada surta efecto. Si no se pasa un identificador de seguridad de Windows (SID de NT) al OAM para un ID de usuario determinado, se intenta obtener el SID adecuado buscando en las bases de datos de seguridad pertinentes.

NTSIDsRequired
Requiere que se pase un SID de NT al OAM al realizar comprobaciones de seguridad.

Para obtener más información, consulte Identificadores de seguridad (SID) deWindows.

Linux AIX **SecurityPolicy = usuario | grupo |UserExternal | por defecto**
En sistemas AIX and Linux , el valor especifica si el gestor de colas utiliza autorización basada en usuario o basada en grupo. Los valores no son sensibles a mayúsculas y minúsculas.

El valor puede ser uno de los siguientes:

grupo
El gestor de colas utiliza la autorización basada en grupo. La autorización para acceder a un recurso se otorga a un grupo.

Un usuario recibe el agregado de todas las autorizaciones que se otorgan a cada grupo al que pertenece.

Los ID de usuario y grupos deben estar definidos en el sistema operativo local.

usuario
El gestor de colas utiliza la autorización basada en el usuario. La autorización para acceder a un recurso se puede otorgar a un grupo o a un ID de usuario específico.

Un usuario recibe el agregado de las autorizaciones siguientes:

- Autorizaciones que se otorgan al usuario específico.
- Autorizaciones que se otorgan a cada grupo al que pertenece el usuario.

Los ID de usuario y grupos deben estar definidos en el sistema operativo local.

UserExternal
El gestor de colas utiliza la autorización basada en el usuario. Sin embargo, se pueden otorgar autorizaciones a los ID de usuario que no son conocidos por el sistema operativo local.

La autorización para acceder a un recurso se puede otorgar a un grupo o a un ID de usuario específico.

Un usuario recibe el agregado de las autorizaciones siguientes:

- Autorizaciones que se otorgan al usuario específico.
- Autorizaciones que se otorgan a cada grupo al que pertenece el usuario.

Si un usuario no es conocido por el sistema operativo local, se considera que pertenece sólo al grupo nobody. Para obtener más información sobre los grupos, consulte Principales y grupos en AIX, Linux, and Windows. El ID de usuario debe tener una longitud máxima de 12 caracteres y debe ajustarse a las Reglas de denominación de objetos IBM MQ.

Puede modificar los gestores de colas existentes para que utilicen esta opción adicional sin perder la configuración actual.

V 9.4.0 Este es el valor predeterminado si elAuthToken se especifica la estrofa.

valor predeterminado
El gestor de colas utiliza la autorización basada en grupo. El comportamiento es el mismo que para la opción `group` .

Este es el valor predeterminado si elAuthToken la estrofa no está especificada.

Reinicie el administrador de colas para que los cambios en el valor del atributo entren en vigor.

Nota: Linux V 9.4.0 AIX DeIBM MQ 9.4.0 , Si elAuthToken se especifica la estrofa, el valor efectivo de la**SecurityPolicy** El atributo de la estrofa de servicio se establece en`UserExternal` . La autenticación de token no está disponible si**SecurityPolicy** está explícitamente establecido en `Grupo` en la estrofa de Servicio. Si**SecurityPolicy** se establece en `Grupo` , eliminar el**SecurityPolicy** atributo de la sección Servicio y, a continuación, reinicie el gestor de colas. Para obtener más información, consulte “Stanza AuthToken del archivo qm.ini” en la página 124.

SharedBindingsUserId=*tipo-usuario*
El atributo **SharedBindingsUserId** solo se aplica si el servicio especificado es el servicio de autorización predeterminado, es decir, el OAM. El atributo **SharedBindingsUserId** sólo se utiliza en relación con los enlaces compartidos. Este valor le permite especificar si el campo *UserIdentifier* en la estructura *IdentityContext*, de la función MQZ_AUTHENTICATE_USER, es el ID de usuario efectivo o el ID de usuario real.

Para obtener información sobre la función MQZ_AUTHENTICATE_USER, consulte MQZ_AUTHENTICATE_USER - Autenticar usuario.

Los valores posibles son:

Valor predeterminado
El valor del campo *UserIdentifier* está establecido como ID de usuario real.

Real
El valor del campo *UserIdentifier* está establecido como ID de usuario real.

Effective
El valor del campo *UserIdentifier* está establecido como ID de usuario efectivo.

FastpathBindingsUserId=*tipo-usuario*
El atributo **FastpathBindingsUserId** solo se aplica si el servicio especificado es el servicio de autorización predeterminado, es decir, el OAM. El atributo **FastpathBindingsUserId** sólo se utiliza con relación a enlaces de vía de acceso rápida. Este valor le permite especificar si el campo *UserIdentifier* en la estructura *IdentityContext*, de la función MQZ_AUTHENTICATE_USER, es el ID de usuario efectivo o el ID de usuario real.

Para obtener información sobre la función MQZ_AUTHENTICATE_USER, consulte MQZ_AUTHENTICATE_USER - Autenticar usuario.

Los valores posibles son:

Valor predeterminado
El valor del campo *UserIdentifier* está establecido como ID de usuario real.

Real
El valor del campo *UserIdentifier* está establecido como ID de usuario real.

Effective
El valor del campo *UserIdentifier* está establecido como ID de usuario efectivo.

IsolatedBindingsUserId= *user-type*
El atributo **IsolatedBindingsUserId** solo se aplica si el servicio especificado es el servicio de autorización predeterminado, es decir, el OAM. El atributo **IsolatedBindingsUserId** solo se utiliza con relación a enlaces aislados. Este valor le permite especificar si el campo *UserIdentifier* en la estructura *IdentityContext*, de la función MQZ_AUTHENTICATE_USER, es el ID de usuario efectivo o el ID de usuario real.

Para obtener información sobre la función MQZ_AUTHENTICATE_USER, consulte MQZ_AUTHENTICATE_USER - Autenticar usuario.

Los valores posibles son:

Valor predeterminado
El valor del campo *UserIdentifier* está establecido como ID de usuario efectivo.

Real
El valor del campo *UserIdentifier* está establecido como ID de usuario real.

Effective
El valor del campo *UserIdentifier* está establecido como ID de usuario efectivo.

Para obtener más información sobre servicios y componentes instalables, consulte la sección Servicios y componentes instalables para AIX, Linux, and Windows.

Para obtener más información sobre los servicios de seguridad en general, consulte Configuración de la seguridad en sistemas AIX and Linux.

Stanza de ejemplo

```
Service:
   Name=AuthorizationService
   EntryPoints=14
```

Conceptos relacionados
Servicios y componentes instalables para AIX, Linuxy Windows
Referencia relacionada
Servicios y componentes instalables en IBM i

Multi Stanza ServiceComponent del archivo qm.ini

La stanza ServiceComponent especifica información para el componente de servicio. Debe especificar la información de los componentes de servicio cuando añade un servicio instalable nuevo. La sección de servicio de autorización está presente de forma predeterminada y el componente asociado, el OAM, está activo.

Las stanzas **Service** y **ServiceComponent** pueden aparecer en cualquier orden y las claves de stanza bajo las mismas pueden aparecer en cualquier orden. Para cualquiera de estas stanzas, deben estar presentes todas las claves de stanza. Si se duplica una clave de stanza, se utilizará la última.

Durante el inicio, el gestor de colas procesa en turnos las entradas del componente de servicio que hay en el archivo de configuración. A continuación, carga el módulo de componente especificado, invocando el punto de entrada del componente, que debe ser el punto de entrada para la inicialización del componente, y lo pasa al manejador de configuración.

Service=*nombre_servicio*
: El nombre del servicio necesario. Este nombre debe coincidir con el valor especificado en el atributo Name de la información de configuración de Servicio.

Name=*nombre_componente*
: El nombre descriptivo del componente de servicio. Este nombre debe ser exclusivo y contener únicamente caracteres que sean válidos para los nombres de objetos de IBM MQ (por ejemplo, nombres de colas). Este nombre aparece en mensajes de operador generados por el servicio. Por lo tanto, es aconsejable que el nombre empiece por un nombre comercial de la empresa o por cualquier otra serie de caracteres que lo distinga del resto de nombres.

Module=*nombre_módulo*
: El nombre del módulo que contendrá el código para este componente. Debe ser un nombre de vía de acceso completo.

ComponentDataSize=*tamaño*
: El tamaño, en bytes, del área de datos del componente que se pasa al componente en cada llamada. Especifique cero si no se necesitan datos del componente.

Stanza de ejemplo

```
ServiceComponent:
   Service=AuthorizationService
   Name=MQSeries.UNIX.auth.service
   Module=amqzfu
   ComponentDataSize=0
```

Para obtener más ejemplos que muestran una stanza AuthorizationService y sus stanzas ServiceComponent asociadas y una stanza NameService y su stanza ServiceComponent asociada, consulte “Stanza de servicio del archivo qm.ini” en la página 155.

Conceptos relacionados

Servicios y componentes instalables para AIX, Linuxy Windows

Referencia relacionada

“Stanza de servicio del archivo qm.ini” en la página 155
La stanza Service se utiliza para realizar cambios en los servicios instalables. Esta stanza contiene el nombre del servicio y el número de puntos de entrada definidos para el servicio.

Servicios y componentes instalables en IBM i

Información de referencia de servicios instalables

Windows Stanza SPX del archivo qm.ini (solo Windows)

La stanza SPX especifica los parámetros de configuración del protocolo SPX. Estos parámetros alteran temporalmente los atributos predeterminados para los canales.

Utilice la stanza SPX del archivo `qm.ini` para especificar los parámetros de configuración del protocolo SPX.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades del gestor de colas de IBM MQ Explorer SPX .

Socket = 5E86 (valor predeterminado) | *socket_number*
Número de socket de SPX en notación hexadecimal. El valor predeterminado es X'5E86'.

BoardNum = 0 (valor predeterminado) | *número_adaptador*
El número de adaptador de la LAN. El valor predeterminado es el adaptador 0.

KeepAlive=NO|YES
Permite activar o desactivar la función KeepAlive.

KeepAlive=YES hace que SPX compruebe periódicamente si el otro extremo de la conexión sigue estando disponible. En caso contrario se cierra el canal.

Library1= *DLLName1*
Nombre DLL de SPX.

El valor predeterminado es WSOCK32.DLL.

Library2= *DLLName2*
Es el mismo que el valor de LibraryName1 y se utiliza si el código se almacena en dos bibliotecas distintas.

El valor predeterminado es WSOCK32.DLL.

ListenerBacklog=número
Permite alterar temporalmente el número predeterminado de solicitudes pendientes para el escucha de SPX.

Cuando se está recibiendo en SPX, se establece un número máximo de solicitudes de conexión pendientes. Esto puede considerarse una reserva de solicitudes que esperan en el socket de SPX a que el escucha acepte la solicitud. Los valores predeterminados de reserva del escucha se muestran en la Tabla 14 en la página 160.

Tabla 14. Peticiones de conexión pendientes predeterminadas (SPX)

Plataforma	**Valor ListenerBacklog predeterminado**
Servidor de Windows	100
Estación de trabajo de Windows	5

Nota: Algunos sistemas operativos tienen soporte para valores superiores al valor predeterminado indicado. Utilícelo para evitar alcanzar el límite de conexiones.

A su vez, puede que algunos sistemas operativos limiten el tamaño de la reserva de SPX, de modo que la reserva de SPX real sería menor que la solicitada aquí.

Si la reserva alcanza los valores indicados en la Tabla 14 en la página 160, la conexión SPX se rechazará y el canal no podrá iniciarse. En los canales de mensajes, el resultado es que el canal queda en estado RETRY y repite la conexión posteriormente. En conexiones de cliente, este recibe el código de razón MQRC_Q_MGR_NOT_AVAILABLE de `MQCONN` y debería reintentar la conexión más tarde.

Multi Stanza SSL del archivo qm.ini

La stanza SSL se utiliza para configurar los canales TLS en un gestor de colas.

Protocolo de estado de certificado en línea (OCSP)

Un certificado puede contener una extensión AuthorityInfoAccess. Esta extensión especifica un servidor con el que se puede establecer contacto mediante el Protocolo de estado de certificado en línea (OCSP). Para permitir que los canales SSL o TLS del gestor de colas utilicen las extensiones AuthorityInfoAccess, asegúrese de que el servidor OCSP especificado en ellos esté disponible, esté correctamente configurado y sea accesible a través de la red. Para obtener más información, consulte Trabajar con certificados revocados.

CrlDistributionPoint (CDP)

Un certificado puede contener una extensión CrlDistributionPoint. Esta extensión contiene un URL que identifica el protocolo utilizado para descargar una lista de revocación de certificados (CRL) y también el servidor con el que debe establecerse contacto.

Si desea permitir que los canales SSL o TLS del gestor de colas utilicen las extensiones CrlDistributionPoint, asegúrese de que el servidor CDP especificado en ellos esté disponible, esté correctamente configurado y sea accesible a través de la red.

La stanza SSL

Utilice la stanza SSL del archivo `qm.ini` para configurar cómo los canales TLS del gestor de colas intentan utilizar las siguientes instalaciones y cómo reaccionan si se producen problemas al utilizarlos.

En cada uno de los casos siguientes, si el valor suministrado no es uno de los valores válidos listados, se toma el valor predeterminado. No se escriben mensajes de error que mencionen que se ha especificado un valor no válido.

OutboundSNI = CHANNEL | HOSTNAME
Si **OutboundSNI** se establece en CANAL, los clientes con capacidad SNI establecen SNI en el nombre de canal de IBM MQ de destino en el sistema remoto al iniciar una conexión TLS.

Si este atributo se establece en HOSTNAME, los clientes que tengan habilitado SNI establecerán la cabecera SNI en el nombre de host, lo que hará que las solicitudes de conexión saliente reciban el certificado predeterminado del gestor de colas remoto durante el reconocimiento TLS, por lo que no se podrán utilizar los certificados por canal.

Nota: Si se utiliza **OutboundSNI**=HOSTNAME para conectarse a un canal remoto con una etiqueta de certificado configurada, la conexión se rechaza con un MQRC_SSL_INITIALIZATION_ERROR y se imprime un mensaje AMQ9673 en los registros de errores del gestor de colas remoto.

AllowOutboundSNI = YES (valor predeterminado) | NO
Si está habilitado, los clientes que tengan habilitado SNI establecerán SNI en el nombre de canal IBM MQ de destino en el sistema remoto al iniciar una conexión TLS. Si este atributo se establece en NO, los clientes que tengan habilitado SNI no establecerán la cabecera SNI lo que hará que las solicitudes de conexión saliente reciban el certificado predeterminado del gestor de colas remoto durante el reconocimiento TLS, por lo que no se podrán utilizar los certificados por canal.

Atención: Deprecated En IBM MQ 9.3.0 la propiedad **AllowOutboundSNI** está en desuso y solo está disponible para fines de compatibilidad con versiones anteriores.

AllowOutboundSNI establecido en YES proporciona la misma función que **OutboundSNI** establecida en CHANNEL, mientras que **AllowOutboundSNI** establecido en NO proporciona la misma función que **OutboundSNI** establecido en HOSTNAME.

Si los atributos **AllowOutboundSNI** y **OutboundSNI** están presentes en la stanza SSL, el valor de **OutboundSNI** tiene prioridad.

AllowedCipherSpecs=*nombre*|*lista nombres*|ALL
Especifica una lista personalizada de CipherSpecs que están ordenadas y habilitadas para ser utilizadas con canales IBM MQ en multiplataformas.

- Un nombre de CipherSpec único.

- Una lista separada por comas de nombres de CipherSpec de IBM MQ que se van a volver a habilitar.
- El valor especial de ALL, que representa todas las CipherSpecs (no se recomienda).

Nota: No debe seleccionar **ALL** CipherSpecs, ya que esto permite los protocolos SSL 3.0 y TLS 1.0 y un gran número de algoritmos criptográficos débiles.

Para obtener más información, consulte Proporcionar una lista personalizada de CipherSpecs ordenadas y habilitadas en IBM MQ for Multiplatforms en el orden CipherSpec en el reconocimiento TLS.

IBM i ALW **AllowTLSV13 =Y | YES | T | TRUE (valor predeterminado) | N | NO | F | FALSE**

Especifica si un gestor de colas va a poder utilizar CipherSpecs de TLS 1.3.

- Y (valor predeterminado), YES (valor predeterminado), T (valor predeterminado) o TRUE (valor predeterminado): habilita TLS 1.3 , lo que permite al gestor de colas utilizar las CipherSpecsde TLS 1.3 .
- N, NO, F o FALSE: inhabilita TLS 1.3, lo que significa que el gestor de colas no puede utilizar las CipherSpecs de TLS 1.3.

Para obtener más información, consulte Habilitación de CipherSpecs.

CDPCheckExtensions= YES |NO (valor predeterminado)

Especifica si los canales TLS de este gestor de canales intentan comprobar los servidores CDP especificados en las extensiones de certificado CrlDistributionPoint.

- YES: los canales TLS intentan comprobar los servidores CDP para determinar si el certificado digital está revocado.
- NO (valor predeterminado): los canales TLS no intentan comprobar los servidores CDP. Este valor es el valor por omisión.

V 9.4.0 **HTTPSKeyStore= serie**

La serie proporciona la vía de acceso a un repositorio de claves pkcs12 que el gestor de colas puede utilizar como almacén de confianza al crear conexiones https salientes, por ejemplo, a un punto final JWKS.

Este archivo debe estar cifrado e ir acompañado de un archivo 'stash' del mismo nombre, es decir, un archivo con una extensión `.sth` , que se utiliza cuando el gestor de colas necesita acceder a este archivo. De forma predeterminada, si no se especifica este atributo, el gestor de colas busca en el subdirectorio `ssl` de los datos del gestor de colas un archivo denominado `mqdefcer.p12`.

Si se actualiza este atributo, debe reiniciar el gestor de colas para empezar a utilizar el nuevo archivo de repositorio de claves especificado.

Consulte Creación de un repositorio de claves para utilizarlo como almacén de confianza TLS para obtener ayuda sobre la creación de un almacén de confianza basado en el entorno del sistema operativo.

ALW **MinimumRSAKeySize=int**

Especifica el tamaño de clave mínimo que deben tener los certificados RSA para poder ser aceptados durante un reconocimiento TLS. Permite cualquier valor igual a 0 o superior. Toma de forma predeterminada el valor 1, si no se especifica.

OCSPAuthentication=REQUIRED (valor predeterminado) | WARN | OPCIONAL

Especifica la acción que se va a realizar cuando el estado de revocación no se puede determinar desde un servidor OCSP.

Si la comprobación de OCSP está habilitada, un programa del canal TLS intenta establecer contacto con un servidor OCSP.

Si el programa del canal no puede ponerse en contacto con ningún servidor OCSP, o si ningún servidor puede proporcionar el estado de revocación del certificado, se utiliza el parámetro OCSPAuthentication.

- `REQUIRED` (valor predeterminado): Si no se puede determinar el estado de revocación, la conexión se cierra con un error. Este valor es el valor por omisión.
- `WARN`: Si no se puede determinar el estado de revocación, se escribe un mensaje en los registros de errores del gestor de colas, pero la conexión puede continuar.
- `OPTIONAL`: Si no se puede determinar el estado de revocación, se permite que la conexión continúe de forma silenciosa. No se emiten avisos ni errores.

OCSPCheckExtensions = YES (valor predeterminado) | NO

Especifica si los canales TLS de este gestor de colas intentan comprobar los servidores OCSP que se especifican en extensiones de certificado AuthorityInfoAccess.

- YES (valor predeterminado): los canales TLS intentan comprobar los servidores OCSP para determinar si se revoca un certificado digital. Este valor es el valor por omisión.
- NO: los canales TLS no intentan comprobar los servidores OCSP.

ALW **OCSPTimeout= *número***

El número de segundos que se debe esperar un programa de respuesta OCSP al realizar una comprobación de revocación.

A partir de IBM MQ 9.3.0, si se establece un valor de 0, se utiliza el tiempo de espera predeterminado de 30 segundos.

Si no se ha establecido ningún valor, se utiliza el valor predeterminado de IBM MQ de 30 segundos.

SSLHTTPProxyName=*serie*

La serie es el nombre de host o la dirección de red del servidor proxy HTTP que IBM Global Security Kit (GSKit) debe utilizar para las comprobaciones de OCSP. Esta dirección puede ir seguida de un número de puerto opcional, delimitado mediante paréntesis. Si no especifica el número de puerto, se utiliza el puerto HTTP predeterminado, el 80.

AIX Para los clientes de 32 bits en AIX, la dirección de red solo puede ser una dirección IPv4.

En otras plataformas, la dirección de red puede ser una dirección IPv4 o IPv6.

Este atributo puede ser necesario si, por ejemplo, un cortafuegos impide el acceso al URL del programa de respuestas OCSP.

ALW **PeerCertChainValidation=*serie***

La serie puede ser uno de los dos valores siguientes:

- `Usepeerchain` **[Valor predeterminado]**: la cadena de certificados proporcionada por el interlocutor se puede utilizar de puente para cualquier intervalo de cadena de confianza al validar los certificados. Con la excepción del certificado raíz.
- `TruststoreOnly` **[No recomendado]**: solo se utilizarán certificados del almacén de confianza para validar el certificado del interlocutor.

ALW **SSLHTTPConnectTimeout = *número*|0**

El número de segundos que se va a esperar a que una conexión de red se establezca correctamente en un servidor HTTP al realizar una comprobación de revocación.

Si no se establece ningún valor, se utiliza el valor predeterminado de IBM MQ de 0 (desactivado).

Stanza de ejemplo

```
SSL:
```

```
 OutboundSNI=CHANNEL
AllowedCipherSpecs=TLS13 CipherSpec list
AllowTLSV13=Y
CDPCheckExtensions=NO
MinimumRSAKeySize=1
OCSPAuthentication=REQUIRED
OCSPCheckExtensions=YES
OCSPTimeout=30
PeerCertChainValidation=Usepeerchain
SSLHTTPConnectTimeout=0
```

Notas:

- El valor predeterminado para **OutboundSNI** es **Channel**.
- La lista **TLS13 CipherSpec** es una lista de CipherSpecs específicas, no los cifrados de alias. Si sólo necesita cifrados TLS1.3 , debe listarlos. Por ejemplo:

  TLS_CHACHA20_POLY1305_SHA256
  TLS_AES_256_GCM_SHA384
  TLS_AES_128_GCM_SHA256
  TLS_AES_128_CCM_SHA256
  TLS_AES_128_CCM_8_SHA256
- El valor predeterminado para **AllowTLSV13** es Y a menos que haya habilitado cifrados débiles, en cuyo caso se desactiva (a menos que lo active explícitamente).
- Los valores de **CDPCheckExtensions** solo pueden ser Sí o No.
- Los valores de **PeerCertChainValidation** solo pueden ser Usepeerchain o Truststoreonly.

Multi Stanza de subagrupación del archivo qm.ini

Esta stanza la crea IBM MQ. No la cambie.

La stanza de subagrupación y el atributo **ShortSubpoolName** dentro de esta stanza se escriben automáticamente mediante IBM MQ donde se crea un gestor de colas. IBM MQ elige un valor para **ShortSubpoolName**. No modifique dicho valor.

El nombre corresponde a un directorio y enlace simbólico creados dentro del directorio /var/mqm/sockets , que IBM MQ utiliza para las comunicaciones internas entre sus procesos en ejecución.

Multi Stanza TCP del archivo qm.ini

La stanza TCP especifica los parámetros de configuración de Protocolo de control de transmisiones/Internet Protocol (TCP/IP). Estos parámetros alteran temporalmente los atributos predeterminados para los canales.

Utilice la stanza TCP del archivo qm.ini para especificar los parámetros de configuración de TCP/IP.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la página de propiedades del gestor de colas TCP de IBM MQ Explorer SPX .

Port = 1414 (valor predeterminado) | *número_puerto*
El número de puerto predeterminado, en notación decimal, para sesiones TCP/IP. El número de puerto *habitual* para IBM MQ es 1414.

Windows **Library1= *NombreDLL1* (sólo Windows)**
El nombre de la DLL de TCP/IP.

El valor predeterminado es WSOCK32.

Multi **SecureCommsSólo = NO | N | FALSE | F (valor predeterminado) |TRUE | T | YES | Y**
Especifique si la comunicación de texto sin formato está permitida o no.

SecureCommsOnly=NO|N|FALSE|F
Se permite la comunicación de texto sin formato y se genera un mensaje de aviso cuando se inicia el gestor de colas.

SecureCommsOnly=YES|Y|TRUE|T
Se permite la comunicación de texto sin formato y se genera un mensaje de información cuando se inicia el gestor de colas.

KeepAlive = NO (valor predeterminado) |SÍ
Permite activar o desactivar la función KeepAlive. KeepAlive=YES hace que TCP/IP compruebe periódicamente si el otro extremo de la conexión sigue estando disponible. En caso contrario se cierra el canal.

ListenerBacklog=número
Permite alterar temporalmente el número predeterminado de peticiones pendientes para el escucha de TCP/IP.

Cuando se está recibiendo en TCP/IP, se define un número máximo de solicitudes de conexión pendientes. Esto puede considerarse una reserva de solicitudes que esperan en el puerto TCP/IP a que el escucha acepte la solicitud. Los valores predeterminados de reserva del escucha se muestran en la Tabla 15 en la página 165.

Tabla 15. Peticiones de conexión pendientes predeterminadas (TCP)

Plataforma	**Valor ListenerBacklog predeterminado**
Windows Windows Servidor	100
Linux Linux	100
AIX AIX V5.3 o posterior	100

Nota: Algunos sistemas operativos tienen soporte para valores superiores al valor predeterminado indicado. Utilícelo para evitar alcanzar el límite de conexiones.

A su vez, puede que algunos sistemas operativos limiten el tamaño de la reserva de TCP, de modo que la reserva de TCP real sería menor que la solicitada aquí.

Si la reserva alcanza los valores indicados en la Tabla 15 en la página 165, la conexión TCP/IP se rechazará y el canal no podrá iniciarse. En los canales de mensajes, el resultado es que el canal queda en estado RETRY y repite la conexión posteriormente. En conexiones de cliente, este recibe el código de razón MQRC_Q_MGR_NOT_AVAILABLE de `MQCONN` y reintenta la conexión más tarde.

El siguiente grupo de propiedades se puede utilizar para controlar el tamaño de los almacenamientos intermedios utilizados por el TCP/IP. Los valores se pasan directamente a la capa del TCP/IP del sistema operativo. Se debe ir con sumo cuidado al utilizar estas propiedades. Una configuración incorrecta de dichos valores, puede afectar negativamente al rendimiento de TCP/IP. Si desea más información sobre cómo puede afectar al rendimiento, consulte la documentación de TCP/IP de su entorno. Un valor de cero indica que el sistema operativo gestionará los tamaños de almacenamiento intermedio, en lugar de que IBM MQ fije los tamaños de almacenamiento intermedio.

Connect_Timeout = 0 (valor predeterminado) |número
El número de segundos antes de que un intento de conectar el socket sobrepase el tiempo de espera. El valor predeterminado de cero especifica que no hay tiempo de espera de conexión.

Los procesos de canal de IBM MQ se conectan a través de sockets no de bloqueo. Por lo tanto, si el otro extremo del socket no está preparado, connect() se devuelve inmediatamente con *EINPROGRESS* o *EWOULDBLOCK*. Después de esto, se intentará de nuevo la conexión, hasta un total de 20 intentos, cuando se notifica un error de comunicación.

Si `Connect_Timeout` se establece en un valor distinto de cero, IBM MQ espera durante el período estipulado durante la llamada select() a que el socket esté preparado. Esto aumenta las posibilidades

de éxito de una llamada connect() posterior. Esta opción podría ser conveniente en situaciones en las que las conexiones requerirían algún período de espera, debido a una gran carga en la red.

SndBuffSize = número |0 (valor predeterminado)
El tamaño en bytes del almacenamiento intermedio de envío TCP/IP que utiliza el extremo emisor de los canales. Este valor de stanza puede ser alterado temporalmente por una stanza más específica para el tipo de canal, por ejemplo RcvSndBuffSize. Si el valor se establece en cero, se utilizan los valores predeterminados de sistema operativo. Si no se establece ningún valor, se utiliza el valor predeterminado de IBM MQ, 32768.

Multi Los nuevos gestores de colas se crean automáticamente con un valor predeterminado de 0 (consulte “Stanza de ejemplo” en la página 166.

RcvBuffSize = número |0 (valor predeterminado)
El tamaño en bytes del almacenamiento intermedio de recepción TCP/IP que utiliza el extremo receptor de los canales. Este valor de stanza puede ser alterado temporalmente por una stanza más específica para el tipo de canal, por ejemplo RcvRcvBuffSize. Si el valor se establece en cero, se utilizan los valores predeterminados de sistema operativo. Si no se establece ningún valor, se utiliza el valor predeterminado de IBM MQ, 32768.

Multi Los nuevos gestores de colas se crean automáticamente con un valor predeterminado de 0 (consulte “Stanza de ejemplo” en la página 166.

RcvSndBuffSize = número |0 (valor predeterminado)
Tamaño en bytes del almacenamiento intermedio de envío TCP/IP que utiliza el extremo emisor de un canal receptor. Si el valor se establece en cero, se utilizan los valores predeterminados de sistema operativo. Si no se establece ningún valor, se utiliza el valor predeterminado de IBM MQ, 32768.

Multi Los nuevos gestores de colas se crean automáticamente con un valor predeterminado de 0 (consulte “Stanza de ejemplo” en la página 166.

RcvRcvBuffSize = número |0 (valor predeterminado)
Tamaño en bytes del almacenamiento intermedio de recepción TCP/IP que utiliza el extremo receptor de un canal receptor. Si el valor se establece en cero, se utilizan los valores predeterminados de sistema operativo. Si no se establece ningún valor, se utiliza el valor predeterminado de IBM MQ, 32768.

Multi Los nuevos gestores de colas se crean automáticamente con un valor predeterminado de 0 (consulte “Stanza de ejemplo” en la página 166.

SvrSndBuffSize = número |0 (valor predeterminado)
Tamaño en bytes del almacenamiento intermedio de envío TCP/IP utilizado por el extremo de servidor de un canal de conexión de cliente y de servidor. Si el valor se establece en cero, se utilizan los valores predeterminados de sistema operativo. Si no se establece ningún valor, se utiliza el valor predeterminado de IBM MQ, 32768.

Multi Los nuevos gestores de colas se crean automáticamente con un valor predeterminado de 0 (consulte “Stanza de ejemplo” en la página 166.

SvrRcvBuffSize = número |0 (valor predeterminado)
Tamaño en bytes del almacenamiento intermedio de recepción TCP/IP utilizado por el extremo de servidor del canal de conexión de cliente y de servidor. Si el valor se establece en cero, se utilizan los valores predeterminados de sistema operativo. Si no se establece ningún valor, se utiliza el valor predeterminado de IBM MQ, 32768.

Multi Los nuevos gestores de colas se crean automáticamente con un valor predeterminado de 0 (consulte “Stanza de ejemplo” en la página 166).

Stanza de ejemplo

```
TCP:
  SndBuffSize=0
```

```
    RcvBuffSize=0
    RcvSndBuffSize=0
    RcvRcvBuffSize=0
    ClntSndBuffSize=0
    ClntRcvBuffSize=0
    SvrSndBuffSize=0
    SvrRcvBuffSize=0
```

Nota: Multi Para los nuevos gestores de colas en Multiplatforms, los tamaños de almacenamiento intermedio de envío y recepción TCP predeterminados en la stanza TCP de `qm.ini file` se establecen para que los gestione el sistema operativo. Tal como se muestra en el ejemplo anterior, los nuevos gestores de colas se crean automáticamente con un valor predeterminado de 0 para los almacenamientos intermedios de envío y recepción. Esto sólo se aplica a los nuevos gestores de colas. Se conservan los valores de almacenamiento intermedio de envío y recepción de TCP para los gestores de colas que se migran de versiones anteriores de IBM MQ.

Si las propiedades de tamaño de almacenamiento intermedio TCP se eliminan del archivo `qm.ini` , el almacenamiento intermedio predeterminado se establece en 32K. Debe tener cuidado al utilizar este valor predeterminado ya que es posible que 32K no sea un almacenamiento intermedio adecuado para todos los escenarios de mensajería.

Si las propiedades de almacenamiento intermedio de envío y recepción de TCP se establecen en cero, se utilizan los valores predeterminados del sistema operativo. El método para elegir estos valores predeterminados variará según el sistema operativo, pero normalmente se puede encontrar en las páginas de manual del sistema operativo "tcp" o get/setsockopt ().

Multi Stanza `TuningParameters` del archivo qm.ini

La stanza TuningParameters especifica opciones para ajustar el gestor de colas.

SuppressDspAuthFail= YES |NO (valor predeterminado)
Cuando se establece en YES, el gestor de colas suprime la generación de sucesos de autorización y la grabación de mensajes de error de AMQ8077 en el registro de errores cuando falla una comprobación de autorización, si la conexión carece de autorización + dsp sobre un objeto.

ImplSyncOpenOutput=*valor*
`ImplSyncOpenOutput` es el número mínimo de aplicaciones que tienen abierta la cola para colocación, antes de que se pueda habilitar un punto de sincronismo implícito para una colocación persistente, fuera del punto de sincronismo. El valor predeterminado de **`ImplSyncOpenOutput`** es 2.

Esto tiene como efecto que, si hay solo una aplicación que tiene esa cola abierta para una operación de colocación, **`ImplSyncOpenOutput`** se desactiva.

Especificar `ImplSyncOpenOutput=1` significa que siempre se contempla un punto de sincronización implícito. Puede configurar cualquier valor entero positivo. Si no desea que se añada nunca un punto de sincronización implícito, establezca `ImplSyncOpenOutput=OFF`.

UniformClusterName=*nombre del clúster*
El nombre del clúster de IBM MQ que está utilizando como un clúster uniforme.

OAMLdapConnectTimeout=`hora|0` (valor predeterminado)
El tiempo máximo, en segundos, que el cliente LDAP esperará para establecer una conexión TCP con el servidor. Si se suministran varios servidores LDAP a través de una lista de nombres de conexión, el tiempo de espera se aplica a cada intento de conexión individual, por lo que se intenta una conexión con la siguiente entrada de la lista de nombres si se alcanza este tiempo de espera.

`time` tiene un valor máximo de 3600 segundos y un valor de `0`, que es el valor mínimo y el valor predeterminado, significa que la espera es ilimitada.

OAMLdapQueryTimeLimit=`time|0` (valor predeterminado)
El tiempo máximo, en segundos, que el cliente LDAP esperará para recibir una respuesta a una solicitud LDAP del servidor, una vez que se haya establecido una conexión y se haya enviado una solicitud LDAP.

`time` tiene un valor máximo de 3600 segundos y un valor de `0`, que es el valor mínimo y el valor predeterminado, significa que la espera es ilimitada.

OAMLdapResponseWarningTime=*umbral*
Si una conexión con un servidor LDAP ha tardado más tiempo que el número de segundos de umbral especificado por el parámetro **`OAMLdapResponseWarningTime`** , se grabará un mensaje AMQ5544W en el registro de errores. El umbral predeterminado es de 10 segundos.

ExpiryInterval
Indica la frecuencia con la que el gestor de colas explora las colas en busca de mensajes caducados que otras actividades de cola todavía no han limpiado. Es un intervalo de tiempo en segundos.

De forma predeterminada, el explorador de caducidad se ejecuta aproximadamente cada cinco minutos en compilaciones IBM MQ de producción.

PRECAUCIÓN: Normalmente no es necesario modificar el valor de **`ExpiryInterval`** , y debe modificar este valor sólo bajo la guía del soporte de IBM .

LivenessHeartBeatLen
Configura la frecuencia con la que el gestor de colas comprueba que las grabaciones en el registro se realizan a una velocidad razonable. El valor máximo para **`LivenessHeartBeatLen`** es de 600 segundos (10 minutos) y el valor mínimo es 0, lo que tiene el efecto de inhabilitar la comprobación por completo.

PRECAUCIÓN: En la mayoría de los casos, no es necesario cambiar la frecuencia de estas comprobaciones. No realice ningún cambio a menos que se lo indique el servicio de soporte de IBM .

ECHeartBeatLen
Configura la frecuencia de las comprobaciones de estado generales del gestor de colas. El valor mínimo para **`ECHeartBeatLen`** es 10000 milisegundos (10 segundos) y el valor máximo es 60000 milisegundos (60 segundos).

PRECAUCIÓN: En la mayoría de los casos, no es necesario cambiar la frecuencia de estas comprobaciones. No realice ningún cambio a menos que se lo indique el servicio de soporte de IBM .

FileLockHeartBeatLen
Cambia el valor predeterminado para las comprobaciones de bloqueo de archivo para un gestor de colas de varias instancias que el controlador de ejecución realiza periódicamente para asegurarse de que todavía mantiene el bloqueo exclusivo en el archivo de varias instancias primario. De forma predeterminada, estas comprobaciones de bloqueo de archivo se realizan cada 20 segundos. El valor mínimo para **`FileLockHeartBeatLen`** es de 10 segundos y el valor máximo es de 600 segundos (10 minutos).

PRECAUCIÓN: En la mayoría de los casos, no es necesario cambiar la frecuencia de estas comprobaciones. No realice ningún cambio a menos que se lo indique el servicio de soporte de IBM .

Stanza de ejemplo

```
TuningParameters:
       SuppressDspAuthFail=NO
       ImplSyncOpenOutput=2
       OAMLdapConnectTimeout=60
       OAMLdapQueryTimeLimit=60
       OAMLdapResponseWarningTime=10
       ExpiryInterval=300
```

Conceptos relacionados
Punto de sincronismo implícito

Multi Stanza Variables del archivo qm.ini

La stanza Variables especifica variables de configuración para utilizarlas con clústeres uniformes automáticos.

Puede utilizar los atributos enumerados en la stanza Variables durante la configuración automática de clúster de CONNAME y los campos de mandato de script de IBM MQ de nombre de canal de un canal de clúster receptor. Las variables de configuración no se pueden utilizar en ningún otro elemento de un script MQSC.

atributo=valor
Especifica un nombre y un valor asociado para utilizarlo como una inserción durante las definiciones de mandato de script de IBM MQ.

Los pares *atributo=valor* se pueden proporcionar utilizando la opción de línea de mandatos **-iv** en el mandato **crtmqm** al crear un gestor de colas.

Stanza de ejemplo

```
Variables:
       CONNAME=127.0.0.1(1414)
```

Conceptos relacionados

"Equilibrio de aplicaciones automático" en la página 434
El equilibrado automático de aplicaciones mejora considerablemente la distribución y la disponibilidad de las aplicaciones habilitando un clúster uniforme de IBM MQ para gestionar de cerca la distribución de aplicaciones en todo el clúster, en lugar de depender de la aleatorización o de un anclaje manual de aplicaciones a gestores de colas específicos.

Tareas relacionadas

"Creación de un nuevo clúster uniforme" en la página 448
Cómo crear un nuevo clúster uniforme.

Referencia relacionada

"Utilización de la configuración de clúster automático" en la página 452
Puede configurar IBM MQ para habilitar la configuración automática cambiando la información de configuración de `qm.ini`.

Multi Stanza XAResourceManager del archivo qm.ini

La stanza XAResourceManager especifica información sobre los gestores de recursos implicados en unidades de trabajo globales coordinadas por el gestor de colas.

Utilice la stanza XAResourceManager del archivo `qm.ini` para especificar la información sobre los gestores de recursos implicados en las unidades de trabajo globales coordinadas por el gestor de colas.

Windows Linux De forma alternativa, en Linux (x86 y x86-64) y Windows, utilice la IBM MQ Explorer página de propiedades del gestor de colas del gestor de recursos XA.

Añada manualmente la información de configuración del gestor de recursos XA para cada instancia de un gestor de recursos que participe en unidades de trabajo globales; no se proporcionan valores predeterminados.

Consulte el apartado Coordinación de bases de datos para obtener más información sobre los atributos de los gestores de recursos.

Name=*nombre* (obligatorio)
Este atributo identifica la instancia del gestor de recursos.

El valor Name puede tener hasta 31 caracteres de largo. Puede utilizar el nombre del gestor de recursos que se ha definido en la estructura de conmutación XA. No obstante, si está utilizando más de una instancia del mismo gestor de recursos, debe crear un nombre exclusivo para cada instancia. Puede asegurar su exclusividad incluyendo el nombre de la base de datos en la serie Name, por ejemplo.

IBM MQ utiliza el valor Name en los mensajes y en la salida del mandato `dspmqtrn`.

No cambie el nombre de una instancia del gestor de recursos ni suprima su entrada de la información de configuración después de que se haya iniciado el gestor de colas asociado y el nombre del gestor de recursos esté en vigor.

SwitchFile=*nombre* (obligatorio)
El nombre completo del archivo de carga que contiene la estructura de conmutación XA del gestor de recursos.

Si utiliza un gestor de colas de 64 bits con aplicaciones de 32 bits, el valor name debe contener únicamente el nombre base del archivo de carga que contiene la estructura de conmutación XA del gestor de recursos.

El archivo de 32 bits se cargará en la aplicación desde la vía de acceso que especifique `ExitsDefaultPath`.

El archivo de 64 bits se cargará en el gestor de colas desde la vía de acceso que especifique `ExitsDefaultPath64`.

XAOpenString=*serie* (opcional)
La serie de datos que se ha de pasar al punto de entrada xa_open del gestor de recursos. El contenido de la serie depende del gestor de recursos propiamente dicho. Por ejemplo, la serie puede identificar la base de datos a la que debe acceder esta instancia del gestor de recursos. Para obtener más información acerca de la definición de este atributo, consulte:

- Añadir información de configuración del gestor de recursos para Db2
- Añadir información de configuración del gestor de recursos para Oracle
- Adición de información de configuración del gestor de recursos para Sybase
- Añadir información de configuración del gestor de recursos para Informix

y, en la documentación del gestor de recursos, la serie de caracteres apropiada.

XACloseString=*serie* (opcional)
La serie de datos que se ha de pasar al punto de entrada xa_close del gestor de recursos. El contenido de la serie depende del gestor de recursos propiamente dicho. Para obtener más información acerca de la definición de este atributo, consulte:

- Añadir información de configuración del gestor de recursos para Db2
- Añadir información de configuración del gestor de recursos para Oracle
- Adición de información de configuración del gestor de recursos para Sybase
- Añadir información de configuración del gestor de recursos para Informix

y, en la documentación de la base de datos, la serie de caracteres adecuada.

ThreadOfControl=THREAD|PROCESS

Windows Este atributo es obligatorio para Windows. El gestor de colas utiliza este valor para la serialización cuando necesita llamar al gestor de recursos desde alguno de sus propios procesos multihebra.

THREAD
El gestor de recursos está totalmente *preparado para hebras*. En un proceso IBM MQ multihebra, se pueden realizar llamadas a función de XA al gestor de recursos externo desde múltiples hebras a la vez.

PROCESS
El gestor de recursos no está *preparado para funcionar con varias hebras*. En un proceso IBM MQ multihebra, sólo se puede realizar una llamada a función de XA a la vez al gestor de recursos.

La entrada **ThreadOfControl** no se aplica a llamadas de función XA emitidas por el gestor de colas en un proceso de aplicaciones multihebra. En general, una aplicación que tiene unidades de trabajo simultáneas en distintas hebras requiere que esta modalidad de operación esté soportada por todos los gestores de colas.

Stanza de ejemplo

```
XAResourceManager:
   Name=DB2 Resource Manager Bank
   SwitchFile=/usr/bin/db2swit
   XAOpenString=MQBankDB
   XACloseString=
   ThreadOfControl=THREAD
```

Nota: El número máximo de stanzas XAResourceManager está limitado a 255. Sin embargo, sólo debería utilizar un número reducido de stanzas para evitar la degradación del rendimiento de las transacciones.

IBM i Archivo qm.ini de ejemplo para IBM i

Un ejemplo que muestra cómo se pueden organizar los grupos de atributos en un archivo de configuración del gestor de colas para IBM i.

```
#*******************************************************************#
#* Module Name: qm.ini                                             *#
#* Type       : IBM MQ queue manager configuration file            *#
# Function    : Define the configuration of a single queue manager *#
#*                                                                 *#
#*******************************************************************#
#* Notes      :                                                    *#
#* 1) This file defines the configuration of the queue manager     *#
#*                                                                 *#
#*******************************************************************#
Log:
LogPath=QMSATURN.Q
LogReceiverSize=65536

CHANNELS:
MaxChannels = 20       ; Maximum number of channels allowed.
                       ; Default is 100.
MaxActiveChannels = 10 ; Maximum number of channels allowed to be
                       ; active at any time. The default is the
                       ; value of MaxChannels.

TCP:                   ; TCP/IP entries.
KeepAlive = Yes        ; Switch KeepAlive on.
SvrSndBuffSize=20000   ; Size in bytes of the TCP/IP send buffer for each
                       ; channel instance. Default is 32768.
SvrRcvBuffSize=20000   ; Size in bytes of the TCP/IP receive buffer for each
                       ; channel instance. Default is 32768.
Connect_Timeout=10000  ; Number of seconds before an attempt to connect the
                       ; channel instance times out. Default is zero (no timeout).

QMErrorLog:
ErrorLogSize = 262144
ExcludeMessage = 7234
SuppressMessage = 9001,9002,9202
SuppressInterval = 30

TuningParameters:
   ImplSyncOpenOutput=2
```

ALW Archivo de configuración de instalación, mqinst.ini

En sistemas AIX and Linux , el archivo de configuración de instalación, `mqinst.ini`, contiene información sobre todas las instalaciones de IBM MQ . En sistemas Windows , la información de configuración de instalación se encuentra en el registro.

Ubicación del archivo `mqinst.ini`

Linux AIX

El archivo `mqinst.ini` se encuentra en el directorio `/etc/opt/mqm` en sistemas AIX and Linux. Contiene información sobre qué instalación, si la hay, es la instalación primaria, así como la siguiente información para cada instalación:

- El nombre de la instalación.
- La descripción de la instalación
- El identificador de la instalación
- La vía de instalación

Importante: El archivo `mqinst.ini` no se debe editar ni referenciar directamente ya que su formato no es fijo y podría cambiar.

El identificador de instalación, para uso interno solamente, se establece automáticamente y no se debe modificar.

En lugar de editar el archivo `mqinst.ini` directamente, debe utilizar los mandatos siguientes para crear, suprimir, consultar y modificar los valores del archivo:

crtmqinst para crear entradas.
dltmqinst para suprimir entradas.
dspmqinst para mostrar entradas.
setmqinst para definir entradas.

Información de configuración de la instalación en Windows

Windows

No hay ningún archivo `mqinst.ini` en Windows. La información de configuración de la instalación está en el registro, en la clave siguiente:

```
HKLM\SOFTWARE\IBM\WebSphere MQ\Installation\InstallationName
```

Importante: Esta clave no debe editarse ni hacer referencia directamente al mismo ya que su formato no está fijado y podría cambiar.

En su lugar, hay que utilizar los mandatos siguientes para consultar y modificar los valores del registro:

dspmqinst para mostrar entradas.
setmqinst para definir entradas.

En Windows, los mandatos **`crtmqinst`** y **`dltmqinst`** no están disponibles. Los procesos de instalación y desinstalación manejan la creación y la supresión de las entradas necesarias del registro.

Multi Archivo de configuración de IBM MQ MQI client , mqclient.ini

Puede configurar los clientes utilizando atributos en un archivo de texto. Estos atributos se pueden alterar temporalmente con variables de entorno o de otras formas según la plataforma específica.

Puede configurar IBM MQ MQI clients utilizando un archivo de texto, similar al archivo de configuración del gestor de colas, `qm.ini`. El archivo contiene un número de stanzas, y cada una de ellas contiene un número de líneas del formato **`attribute-name`**=*valor*.

El archivo de configuración de IBM MQ MQI client generalmente se denomina `mqclient.ini` pero puede optar por asignarle otro nombre. La información de configuración de este archivo se aplica a las plataformas siguientes:

- ALW AIX, Linux, and Windows
- IBM i IBM i

  Nota: En IBM i, no hay ningún archivo `mqclient.ini` predeterminado. Sin embargo, puede crear el archivo en el IBM i Integrated File System (IFS).

Para obtener más información, consulte “Ubicación del archivo de configuración de cliente” en la página 174.

Nota: z/OS La plataforma z/OS no se puede utilizar para ejecutar clientes IBM MQ . Por lo tanto, el archivo `mqclient.ini` no existe en IBM MQ for z/OS.

Los atributos del archivo de configuración IBM MQ MQI client se aplican a los clientes que utilizan:

- La MQI
- IBM MQ classes for Java
- IBM MQ classes for JMS
- IBM MQ classes for .NET
- XMS

Aunque los atributos el archivo de configuración de IBM MQ MQI client se aplican a la mayoría de clientes de IBM MQ, hay algunos atributos que no son leídos por clientes gestionados de .NET y XMS .NET, ni por clientes que usen IBM MQ classes for Java o IBM MQ classes for JMS. Para obtener más información, consulte “Qué clientes de IBM MQ pueden leer cada atributo” en la página 175.

Las características de configuración se aplican a todas las conexiones que una aplicación cliente establece con cualquier gestor de colas, en lugar de ser específicas de una conexión individual con un gestor de colas. Los atributos relacionados con una conexión con un gestor de colas individual se pueden configurar mediante programación, por ejemplo, utilizando una estructura MQCD o bien utilizando una tabla de definiciones de canal de cliente (CCDT).

A continuación se muestra un ejemplo de archivo de configuración de cliente:

```
#* Module Name: mqclient.ini                                          *#
#* Type       : IBM MQ MQI client configuration file                  *#
#  Function   : Define the configuration of a client                  *#
#*                                                                    *#
#**********************************************************************#
#* Notes      :                                                       *#
#* 1) This file defines the configuration of a client                 *#
#*                                                                    *#
#**********************************************************************#

ClientExitPath:
   ExitsDefaultPath=/var/mqm/exits
   ExitsDefaultPath64=/var/mqm/exits64

TCP:
   Library1=DLLName1
   KeepAlive = Yes
   ClntSndBuffSize=32768
   ClntRcvBuffSize=32768
   Connect_Timeout=0

MessageBuffer:
   MaximumSize=-1
   Updatepercentage=-1
   PurgeTime=0

LU62:
   TPName
   Library1=DLLName1
   Library2=DLLName2
```

```
PreConnect:
    Module=myMod
    Function=myFunc
    Data=ldap://myLDAPServer.com:389/cn=wmq,ou=ibm,ou=com
    Sequence=1

CHANNELS:
DefRecon=YES
      ServerConnectionParms=SALES.SVRCONN/TCP/hostname.x.com(1414)

Connection:
     ApplName=ExampleApplName
```

No se pueden configurar varias conexiones de canal mediante el archivo de configuración de cliente.

Las variables de entorno que estaban soportadas en releases anteriores a IBM WebSphere MQ 7.0 siguen estando soportadas en releases posteriores y cuando una variable de ese entorno coincide con un valor equivalente en el archivo de configuración de cliente, la variable de entorno sustituye el valor del archivo de configuración de cliente.

En una aplicación cliente que utilice IBM MQ classes for JMS, también se puede sustituir el archivo de configuración de cliente de las siguientes maneras:

- Definiendo propiedades en el archivo de configuración de JMS.
- Configurando propiedades de sistema de Java que también sustituyen el archivo de configuración de JMS.

En el cliente de .NET, también se puede sustituir el archivo de configuración de cliente y las variables de entorno equivalentes usando el archivo de configuración de aplicación de .NET.

Comentarios en el archivo de configuración

Linux AIX

Puede utilizar el carácter de punto y coma '`;`' y hash '`#`' para marcar el inicio de un comentario dentro del archivo de configuración. Esto puede marcar una línea completa como un comentario o indicar un comentario al final de una línea que no se incluirá en el valor de un ajuste.

Si un valor requiere cualquiera de estos caracteres, debe escapar ese carácter utilizando el carácter de barra inclinada invertida '\'.

En el ejemplo siguiente se muestra el uso de comentarios en el archivo de configuración:

```
# Example of an SSL stanza with comments
SSL:
    ClientRevocationChecks=REQUIRED ; Example of an end of line comment
    SSLCryptoHardware=GSK_PKCS11=/driver\;label\;password\;SYMMETRIC_CIPHER_ON # Example of
escaped comment characters.
```

Conceptos relacionados

Las clases de IBM MQ para el archivo de configuración Java

Multi Ubicación del archivo de configuración de cliente

Un archivo de configuración de IBM MQ MQI client puede mantenerse en varias ubicaciones.

Una aplicación cliente utiliza la siguiente vía de acceso de búsqueda para localizar el archivo de configuración de IBM MQ MQI client:

1. La ubicación especificada por la variable de entorno **MQCLNTCF**.

   El formato de esta variable de entorno es un URL completo. Esto significa que el nombre de archivo puede no ser necesariamente `mqclient.ini` y facilita colocar el archivo en un sistema de archivos conectado a la red.

   Notas:

- Los clientes C, .NET y XMS sólo dan soporte al protocolo `file:` ; se presupone el protocolo `file:` si la serie de URL no empieza por `protocol:`
- Para permitir los JRE Java 1.4.2 , que no dan soporte a la lectura de variables de entorno, la variable de entorno **MQCLNTCF** se puede alterar temporalmente con una propiedad del sistema **MQCLNTCF** Java .

2. Un archivo llamado `mqclient.ini` en el directorio actual de la aplicación.
3. Un archivo denominado `mqclient.ini` en el directorio de datos de IBM MQ para sistemas AIX, Linux, and Windows.

   Notas:

   - El directorio de datos IBM MQ no existe en los casos siguientes:
     - IBM i En IBM i
     - z/OS En z/OS
     - donde el cliente se ha suministrado con otro producto

   IBM i En IBM i, no hay ningún archivo `mqclient.ini` predeterminado. Sin embargo, el archivo se puede crear en el IBM i Integrated File System (IFS) en el directorio `/QIBM/UserData/mqm/`y en la variable de entorno **MQCLNTCF** definida para que apunte a él. Por ejemplo:

   ```
   ADDENVVAR ENVVAR(MQCLNTCF) VALUE('QIBM/UserData/mqm/mqclient.ini') REPLACE(*YES)
   ```

   Para obtener más ejemplos de variables de entorno, consulte “Descripciones de variables de entorno” en la página 67.

   z/OS La plataforma z/OS no se puede utilizar para ejecutar clientes IBM MQ . Por lo tanto, el archivo `mqclient.ini` no existe en IBM MQ for z/OS.

   - Linux AIX En sistemas AIX and Linux , el directorio es `/var/mqm`.
   - Windows En las plataformas Windows , configure la variable de entorno **MQ_DATA_PATH** durante la instalación para que apunte al directorio de datos. Normalmente es `C:\ProgramData \IBM \MQ.`

     Nota: Si está instalando sólo un cliente, la variable de entorno puede ser **MQ_FILE_PATH**.
   - Para permitir los JRE de Java 1.4.2 que no dan soporte a la lectura de variables de entorno, puede alterar manualmente la variable de entorno **MQ_DATA_PATH** con una propiedad del sistema **MQ_DATA_PATH** Java .

4. Un archivo llamado `mqclient.ini` en un directorio estándar adecuado para la plataforma y accesible para los usuarios:
   - Para todos los clientes de Java, este es el valor de la propiedad del sistema `user.home` Java.
   - Linux AIX Para clientes C en plataformas AIX and Linux , este es el valor de la variable de entorno **HOME** .
   - Windows Para clientes C en Windows , son los valores concatenados de las variables de entorno **HOMEDRIVE** y **HOMEPATH** .

Multi Qué clientes de IBM MQ pueden leer cada atributo

La mayoría de los atributos del archivo de configuración de IBM MQ MQI client pueden ser utilizados por el cliente C y los clientes no gestionados de .NET. Sin embargo, hay algunos atributos que no son leídos por clientes gestionados de .NET ni por clientes que usen XMS .NET o IBM MQ classes for Java o el IBM MQ classes for JMS.

Tabla 16. Qué atributos se aplican a cada tipo de cliente

Nombre de stanza y atributos de `mqclient.ini`	**Descripción**	**C y no gestionados de .NET**	**Java**	**JMS**	**.NET gestionado**	**XMS .NET gestionado**
stanza CHANNELS						
CCSID	Juego de caracteres codificado que va a usarse.	Sí	No	No	Sí	Sí
ChannelDefinitionDirectory	Vía de acceso de directorio al archivo que contiene la tabla de definiciones de canal de cliente.	Sí	No	No	Sí	Sí
ChannelDefinitionFile	Nombre del archivo que contiene la tabla de definiciones de canal de cliente.	Sí	No	No	Sí	Sí
ReconDelay	Opción administrativa para configurar el retardo de reconexión de los programas cliente que pueden reconectarse de forma automática.	Sí	No	Sí	Sí	Sí

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	**Descripción**	**C y no gestionados de .NET**	**Java**	**JMS**	**.NET gestionado**	**XMS .NET gestionado**
DefRecon	Opción administrativa para habilitar los programas cliente a fin de que puedan reconectarse de forma automática o para inhabilitar la reconexión automática de un programa cliente que se ha desarrollado para reconectarse de forma automática.	Sí	No	Sí	Sí	Sí
MQReconnectTimeout	El tiempo de espera excedido en segundos para volver a conectar a un cliente.	Sí	No	No	Sí	No
ServerConnectionParms	Ubicación del servidor de IBM MQ y el método de comunicación que hay que usar.	Sí	No	No	Sí	Sí
Put1DefaultAlwaysSync	Controla el comportamiento de la llamada a la función MQPUT1 con la opción `MQPMO_RESPONSE_AS_Q_DEF`.	Sí	Sí	Sí	Sí	Sí

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	Descripción	C y no gestionados de .NET	Java	JMS	.NET gestionado	XMS .NET gestionado
PasswordProtection	Permite definir contraseñas protegidas en la estructura MQCSP en lugar de utilizar SSL o TLS.	Sí	Sí	Sí	Sí	Sí
Stanza ClientExitPath						
ExitsDefaultPath	Especifica la ubicación de las salidas de canal de 32 bits para clientes.	Sí	Sí	Sí	Sí	Sí
ExitsDefaultPath64	Especifica la ubicación de las salidas de canal de 64 bits para clientes.	Sí	Sí	Sí	Sí	Sí
JavaExitsClassPath	Los valores que deben añadirse a la vía de acceso de clases cuando se ejecuta una salida de Java.	No	Sí	Sí	No	No
Stanza Connection						
ApplName	El nombre de aplicación especificado en el archivo de configuración de cliente.	Sí	No	No	No	No
Stanza JMQI						

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	**Descripción**	**C y no gestionados de .NET**	**Java**	**JMS**	**.NET gestionado**	**XMS .NET gestionado**
useMQCSPauthentication	Controla si las aplicaciones IBM MQ classes for Java y IBM MQ classes for JMS tiene que usar el modo de compatibilidad o el modo de autenticación MQCSP al autenticarse con un gestor de colas.	No	Sí	Sí	No	No
Stanza MessageBuffer						
MaximumSize	Tamaño, en kilobytes, del almacenamiento intermedio de lectura anticipada, en el intervalo 1 a 999.999.	Sí	Sí	Sí	Sí	Sí
PurgeTime	Intervalo, en segundos, tras el cual se purgan los mensajes dejados en el almacenamiento intermedio de lectura anticipada.	Sí	Sí	Sí	Sí	Sí

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	**Descripción**	**C y no gestionados de .NET**	**Java**	**JMS**	**.NET gestionado**	**XMS .NET gestionado**
UpdatePercentage	El valor del porcentaje de actualización, en el intervalo 1 - 100, utilizado para calcular el valor de umbral a determinar cuando una aplicación cliente realiza una solicitud nueva al servidor.	Sí	Sí	Sí	Sí	Sí
Stanza PreConnect						
Datos	URL del depósito en el que se almacenan las definiciones de conexión.	Sí	No	No	No	No
Función	Nombre del punto de entrada funcional a la biblioteca que contiene el código de salida de Preconnect.	Sí	No	No	No	No
Módulo	El nombre del módulo que contiene el código de salida de la API.	Sí	No	No	No	No

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	**Descripción**	**C y no gestionados de .NET**	**Java**	**JMS**	**.NET gestionado**	**XMS .NET gestionado**
Sequence	La secuencia en la que se llama a esta salida en relación con otras salidas.	Sí	No	No	No	No
Stanza de seguridad						
DisableClientAMS	Inhabilita o habilita AMS para las conexiones de cliente a un gestor de colas.	Sí	Sí	Sí	No	No
Stanza SSL						
OutboundSNI	Especifica si los clientes que tengan habilitado SNI establecerán SNI en el nombre de canal IBM MQ de destino en el sistema remoto al iniciar una conexión TLS o al nombre de host.	Sí	Sí	Sí	Sí	No

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	Descripción	C y no gestionados de .NET	Java	JMS	.NET gestionado	XMS .NET gestionado
AllowOutboundSNI	Especifica si los clientes que tengan habilitado SNI establecerán SNI en el nombre de canal IBM MQ de destino en el sistema remoto al iniciar una conexión TLS. **Atención:** Deprecated A partir de IBM MQ 9.3.0 , esta propiedad está en desuso. En su lugar, utilice **OutboundSNI** .	Sí	Sí	Sí	No	No
AllowTLSV13	Indica si un gestor de colas puede utilizar las TLS 1.3 CipherSpecs.	Sí (clientes C/C++)	No	No	No	No

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	**Descripción**	**C y no gestionados de .NET**	**Java**	**JMS**	**.NET gestionado**	**XMS .NET gestionado**
CDPCheckExtensions	Especifica si los canales SSL o TLS de este gestor de colas intentan comprobar los servidores CDP especificados en las extensiones de certificado CrlDistributionPoint.	Sí	No	No	No	No
CertificateLabel	la etiqueta de certificado de la definición de canal.	Sí	No	No	No	No
CertificateValPolicy	Determina el tipo de validación de certificados utilizado.	Sí	No	No	No	No
ClientRevocationChecks	Determina cómo se configura la comprobación de revocación de certificados si la llamada de conexión del cliente utiliza un canal SSL/TLS.	Sí	No	No	No	No

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	Descripción	C y no gestionados de .NET	Java	JMS	.NET gestionado	XMS .NET gestionado
EncryptionPolicySuiteB	Determina si un canal utiliza cifrado compatible con Suite B y qué nivel de potencia se utilizará.	Sí	No	No	No	No
EnvironmentScope	Controla si IBM MQ utiliza un único entorno IBM Global Security Kit (GSKit) para todo el proceso o un entorno GSKit por conexión.	Sí (clientes C)	No	No	No	No
MinimumRSAKeyTamaño	Especifica el tamaño de clave mínimo que deben tener los certificados RSA para poder aceptarse.	Sí (clientes C/C++)	No	No	No	No
OCSPAuthentication	Define el comportamiento de IBM MQ cuando se habilita OCSP y la comprobación de revocación de OCSP no puede determinar el estado de revocación del certificado.	Sí	No	No	No	No

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	Descripción	C y no gestionados de .NET	Java	JMS	.NET gestionado	XMS .NET gestionado
OCSPCheckExtensions	Controla si IBM MQ actúa en las extensiones de certificado AuthorityInfo Access.	Sí	No	No	No	No
OCSPTimeout	El número de segundos que se debe esperar un programa de respuesta OCSP al realizar una comprobación de revocación.	Sí	No	No	No	No
ALW PeerCertChainValidation	El valor de validación de certificado de GSKit.	Sí	No	No	No	No
SSLCryptoHardware	Establece la serie de parámetros necesaria para configurar el hardware de cifrado PKCS #11 existente en el sistema.	Sí	No	No	No	No

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de **mqclient.ini**	Descripción	C y no gestionados de .NET	Java	JMS	.NET gestionado	XMS .NET gestionado
SSLCryptoHardwareKeyFile	Especifica la vía de acceso completa y el nombre del archivo que contiene la clave inicial que se ha utilizado para cifrar la contraseña en la serie de configuración de hardware criptográfico PKCS #11 que se ha especificado con el atributo **SSLCryptoHardware** .	Sí	No	No	No	No
SSLFipsRequired	Especifica si sólo se van a utilizar algoritmos certificados por FIPS si se lleva a cabo el cifrado en IBM MQ.	Sí	No	No	No	No
SSLHTTPProxyName	La serie es el nombre de host o la dirección de red del servidor proxy HTTP que GSKit debe utilizar para las comprobaciones de OCSP.	Sí	No	No	No	No

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	Descripción	C y no gestionados de .NET	Java	JMS	.NET gestionado	XMS .NET gestionado
SSLHTTPConnectTimeout	El número de segundos que se va a esperar a que una conexión de red se establezca correctamente en un servidor HTTP al realizar una comprobación de revocación.	Sí	No	No	No	No
SSLKeyRepository	La ubicación del depósito de claves que contiene el certificado digital del usuario, en formato raíz.	Sí	No	No	No	No
ContraseñadeSSLKeyRepository	Frase de contraseña para acceder al repositorio de claves.	Sí	No	No	No	No
SSLKeyResetCount	El número de bytes no cifrados enviados y recibidos en un canal SSL o TLS antes de que se cambie la clave secreta.	Sí	No	No	No	No
stanza TCP						

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	Descripción	C y no gestionados de .NET	Java	JMS	.NET gestionado	XMS .NET gestionado
ClntRcvBuffSize	El tamaño en bytes del almacenamiento intermedio de recepción TCP/IP utilizado por el extremo del cliente de un canal de conexión de cliente - conexión de servidor.	Sí	Sí	Sí	Sí	Sí
ClntSndBuffSize	El tamaño en bytes del almacenamiento intermedio de envío TCP/IP utilizado por el extremo del cliente del canal de conexión del cliente y del servidor.	Sí	Sí	Sí	Sí	Sí
Connect_Timeout	El número de segundos antes de que un intento de conectar el socket sobrepase el tiempo de espera.	Sí	Sí	Sí	No	No
IPAddressVersion	Especifica el protocolo IP que se tiene que utilizar en una conexión de canal.	Sí	No	No	Sí	Sí

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)

Nombre de stanza y atributos de `mqclient.ini`	**Descripción**	**C y no gestionados de .NET**	**Java**	**JMS**	**.NET gestionado**	**XMS .NET gestionado**
KeepAlive	Activa o desactiva la función KeepAlive.	Sí	Sí	Sí	Sí	Sí
Windows Library1	En Windows solo, nombre de la DDL de sockets TCP/IP.	Sí	No	No	No	No
Stanza Trace **Nota:** La stanza Trace sólo se aplica a los clientes IBM MQ .NET y XMS .NET .						
V 9.4.0 MQDotnetTraceNivel	Se utiliza para habilitar el rastreo de IBM MQ .NET .	No	No	No	Sí	No
V 9.4.0 MQDotnetTraceVía de acceso	Apunta a una carpeta donde se crearán los archivos de rastreo de IBM MQ .NET .	No	No	No	Sí	No
V 9.4.0 MQDotnetErrorVía de acceso	Apunta a una carpeta en la que se crearán archivos de registro de errores para el rastreo de IBM MQ .NET .	No	No	No	Sí	No
V 9.4.0 XMSDotnetTraceNivel	Se utiliza para habilitar el rastreo de XMS .NET .	No	No	No	No	Sí
V 9.4.0 XMSDotnetTraceFilePath	Apunta a una carpeta en la que se crearán los archivos de rastreo de XMS .NET .	No	No	No	No	Sí

Tabla 16. Qué atributos se aplican a cada tipo de cliente (continuación)						
Nombre de stanza y atributos de `mqclient.ini`	**Descripción**	**C y no gestionados de .NET**	**Java**	**JMS**	**.NET gestionado**	**XMS .NET gestionado**
V 9.4.0 EspecificaciónXMSDotnetTrace	Especifica el nombre de la clase que desea rastrear para XMS .NET.	No	No	No	No	Sí
V 9.4.0 EspecificaciónXMSDotnetTrace	Especifica el tamaño máximo del archivo de rastreo que se debe generar para XMS .NET.	No	No	No	No	Sí
V 9.4.0 XMSDotnetTraceFileSize	El número de archivos de rastreo que se van a conservar para XMS .NET.	No	No	No	No	Sí

Stanza `Application` del archivo de configuración del cliente

Utilice la stanza `Application` para especificar atributos que influyan en el comportamiento uniforme de equilibrio de clúster para una aplicación específica que se conecte utilizando esta configuración. Los valores de esta stanza tienen prioridad sobre la stanza `ApplicationDefaults`, pero se pueden alterar temporalmente mediante una estructura MQBNO que se suministra mediante un programa.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

Los atributos siguientes se pueden incluir en la stanza Application:

Name = `ApplicationName`
Identifica a qué nombre de aplicación se aplican las opciones.

Tipo = *Simple,ReqRep*
Indica a IBM MQ el patrón general de actividad de IBM MQ en el que participa esta aplicación.

BalanceTimeout = *Never,Immediate,0-999999999, Valor predeterminado*
Indica a IBM MQ el tiempo de espera antes de que se pueda interrumpir la actividad de la aplicación para permitir la reequilibrio; ya sea nunca, o un valor de hasta un máximo de 999.999.999 segundos, con un valor predeterminado de 10 segundos.

BalanceOptions = *None,IgnTrans*
No se establece ninguna opción de equilibrio o permite la interrupción inmediata de las aplicaciones que participan actualmente en una transacción.

Multi Stanza ApplicationDefaults del archivo de configuración del cliente

Utilice la stanza ApplicationDefaults para especificar atributos que influyen en el comportamiento de equilibrio del clúster uniforme predeterminado para las aplicaciones cliente que se conectan utilizando esta configuración. Estos valores predeterminados se pueden alterar temporalmente mediante una stanza Application específica de la aplicación o una estructura MQBNO proporcionada mediante un programa.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

En la stanza ApplicationDefaults se pueden incluir los atributos siguientes:

Tipo = *Simple,ReqRep*
Indica a IBM MQ el patrón general de actividad de IBM MQ en el que participa esta aplicación.

BalanceTimeout = *Never,Immediate,0-999999999, Valor predeterminado*
Indica a IBM MQ el tiempo de espera antes de que se pueda interrumpir la actividad de la aplicación para permitir la reequilibrio; ya sea nunca, o un valor de hasta un máximo de 999.999.999 segundos, con un valor predeterminado de 10 segundos.

BalanceOptions = *None,IgnTrans*
No se establece ninguna opción de equilibrio o permite la interrupción inmediata de las aplicaciones que participan actualmente en una transacción.

Multi Stanza CHANNELS del archivo de configuración de cliente

Utilice la stanza CHANNELS para especificar información sobre canales de cliente.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

Los siguientes atributos pueden incluirse en la stanza CHANNELS:

CCSID = *número*
Juego de caracteres codificado que va a usarse.

Este atributo puede ser leído por clientes C, .NET no gestionados, .NET gestionados y XMS .NET gestionados.

El número de identificador de juego de caracteres codificados es equivalente a la variable de entorno MQCCSID.

ChannelDefinitionDirectory = *ruta*
Vía de acceso de directorio al archivo que contiene la tabla de definiciones de canal de cliente.

Este atributo puede ser leído por clientes C, .NET no gestionados, .NET gestionados y XMS .NET gestionados.

Windows En sistemas Windows, el valor predeterminado es el directorio de archivos de registro y datos de IBM MQ, normalmente `C:\ProgramData \IBM \MQ`.

Linux AIX En sistemas AIX and Linux, el valor predeterminado es `/var/mqm`.

ChannelDefinitionDirectory puede contener un URL que trabaja en combinación con el atributo ChannelDefinitionFile (consulte “Acceso de URL a la tabla de definición de canal de cliente” en la página 55).

La vía de acceso ChannelDefinitionDirectory es equivalente a la variable de entorno **MQCHLLIB**.

ChannelDefinitionFile = *nombrearchivo*|AMQCLCHL.TAB
Nombre del archivo que contiene la tabla de definiciones de canal de cliente.

Este atributo puede ser leído por clientes C, .NET no gestionados, .NET gestionados y XMS .NET gestionados.

La tabla de definición de canal de cliente es equivalente a la variable de entorno **MQCHLTAB**.

ReconDelay = (delay[,rand]) (delay[,rand])...
El atributo ReconDelay proporciona una opción administrativa para configurar el retardo de reconexión de los programas cliente que pueden reconectarse de forma automática.

Este atributo puede ser leído por clientes C, .NET no gestionados, IBM MQ classes for JMS, .NET gestionados y XMS .NET gestionados.

A continuación, se muestra un ejemplo de configuración:

```
ReconDelay=(1000,200)(2000,200)(4000,1000)
```

El ejemplo que se muestra define un retardo inicial de un segundo, además de un intervalo aleatorio de hasta 200 milisegundos. El retardo siguiente es de dos segundos más un intervalo aleatorio de hasta 200 milisegundos. Todos los retardos siguientes son cuatro segundos, además de un intervalo aleatorio de hasta 1000 milisegundos.

DefRecon = NO|YES|QMGR|DISABLED
El atributo DefRecon proporciona una opción administrativa para habilitar los programas cliente con el fin de que lleven a cabo una reconexión automática o para inhabilitar la reconexión automática de un programa cliente que se ha grabado para realizar la reconexión automáticamente. Puede optar por establecer éste último si un programa utiliza una opción, como por ejemplo MQPMO_LOGICAL_ORDER, que es incompatible con la reconexión.

Este atributo puede ser leído por C, .NET sin gestionar, IBM MQ classes for JMS, .NET gestionado y clientes XMS .NET gestionados.

IBM MQ classes for Java no da soporte a la reconexión automática del cliente.

La reconexión automática de cliente normalmente depende de dos valores que son:

- Opción de reconexión establecida en la aplicación MQCONNX (o fábrica de conexiones JMS)
- Opción de reconexión predeterminada proporcionada en cualquier definición de conexión de cliente en uso (estructura MQCD, por ejemplo proporcionada utilizando un archivo CCDT).

El atributo de archivo `mqclient.ini` se aplica **sólo** si no se utiliza ninguna definición de canal que establezca el atributo **DefReconnect** y en esa situación se comporta como si se hubiera proporcionado una. El atributo **DefReconnect** del canal (y, por lo tanto, este atributo, si procede):

- Alterar temporalmente el código de aplicación si se establece en DISABLED
- Se alteran temporalmente mediante el código de aplicación en todos los demás casos, si las opciones se especifican en MQCONNX

Consulte la descripción de DEFRECON para ver una tabla que muestra todas las combinaciones posibles de valores suministrados de definición de canal y aplicación.

Notas:

- Si un MQCD está en uso pero es anterior a MQCD_VERSION_10, el parámetro **DefReconnnect** no forma parte de la estructura. En esta situación, el valor de ese parámetro que falta se rellena con el valor **DefReconnnect** del archivo `mqclient.ini` si se especifica uno. Esto puede ocurrir, por ejemplo, si una aplicación cliente sigue utilizando una CCDT de formato binario generada en una versión anterior de IBM MQ .
- Cuando lo interpreta el código de cliente IBM MQ , una CCDT JSON, consulte "Configuración de una tabla de definición de canal de cliente en formato JSON" en la página 47, siempre genera estructuras MQCD en la versión más reciente y, por lo tanto, siempre proporciona el valor predeterminado (NO) para este atributo a menos que se presente explícitamente con un valor diferente.

MQReconnectTimeout

El tiempo máximo, en segundos, que la función de reconexión automática de cliente de un cliente intenta volver a establecer la conexión. El valor predeterminado es 1800 segundos (30 minutos).

Este atributo puede ser leído por clientes C y .NET no gestionados y clientes .NET gestionados.

Los clientes IBM MQ classes for JMS pueden especificar un tiempo de espera para reconectarse usando la propiedad de fábrica de conexiones CLIENTRECONNECTTIMEOUT. El valor predeterminado de esta propiedad es 1800 segundos (30 minutos).

Los clientes de IBM MQ classes for XMS .NET pueden especificar un tiempo de espera para reconectarse utilizando las siguientes propiedades:

- La propiedad de fábrica de conexiones CLIENTRECONNECTTIMEOUT. El valor predeterminado de esta propiedad es 1800 segundos (30 minutos). Esta propiedad sólo es válida para la modalidad gestionada.
- La propiedad XMSC.WMQ_CLIENT_RECONNECT_TIMEOUT. El valor predeterminado de esta propiedad es 1800 segundos (30 minutos). Esta propiedad sólo es válida para la modalidad gestionada.

ServerConnectionParms

Los parámetros ServerConnectionson equivalentes a la variable de entorno *MQSERVER* y especifican la ubicación del servidor IBM MQ y el método de comunicación que se va a utilizar.

Este atributo puede ser leído por clientes C, .NET no gestionados, .NET gestionados y XMS .NET gestionados.

El atributo ServerConnectionParms define únicamente un canal simple; no puede utilizarlo para definir un canal TLS o un canal con salidas de canal. Es una serie con formato *NombreCanal*/*TipoTransporte*/*NombreConexión*, donde *NombreConexión* debe ser un nombre de red totalmente calificado. *NombreCanal* contener el carácter de barra inclinada (/) porque este carácter se utiliza para separar el nombre de canal, el tipo de transporte y el nombre de conexión.

Cuando se utiliza ServerConnectionParms para definir un canal de cliente, se utiliza una longitud máxima de mensaje de 100 MB. Por consiguiente, el tamaño máximo de mensaje en vigor para el canal es el valor especificado en el canal SVRCONN en el servidor.

Tenga en cuenta que sólo puede realizarse una única conexión de canal de cliente. Por ejemplo, si tiene dos entradas:

```
ServerConnectionParms=R1.SVRCONN/TCP/localhost(1963)
ServerConnectionParms=R2.SVRCONN/TCP/localhost(1863)
```

sólo se utiliza la segunda.

Especifique *ConnectionName* como una lista separada por comas de nombres para el tipo de transporte indicado. Por lo general, sólo se necesita un nombre. Puede proporcionar varios *nombres de host* para configurar varias conexiones con las mismas propiedades. Las conexiones se intentan en el orden en el que se especifican en la lista de conexiones, hasta que se establece una conexión satisfactoriamente. Si no hay una conexión satisfactoria, el cliente inicia el proceso otra vez. Las listas de conexiones son una alternativa para que los grupos de gestores de colas configuren conexiones para clientes reconectables.

Put1DefaultAlwaysSync = NO (valor predeterminado) | YES

Controla el comportamiento de la llamada a la función MQPUT1 con la opción MQPMO_RESPONSE_AS_Q_DEF.

Este atributo puede ser leído por clientes C, .NET no gestionados, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gestionados y XMS .NET gestionados.

NO
Si MQPUT1 se establece con MQPMO_SYNCPOINT, se comporta como MQPMO_ASYNC_RESPONSE. De forma similar, si MQPUT1 se establece con MQPMO_NO_SYNCPOINT, se comporta como MQPMO_SYNC_RESPONSE. Éste es el valor predeterminado.

SÍ
MQPUT1 se comporta como si se ha establecido MQPMO_SYNC_RESPONSE, independientemente de si se ha establecido MQPMO_NO_SYNCPOINT o MQPMO_SYNCPOINT.

PasswordProtection = Compatible (valor predeterminado) |always|opcional

A partir de IBM MQ 8.0, las credenciales de autenticación que las aplicaciones IBM MQ client especifican cuando se conectan a un gestor de colas se pueden proteger utilizando la característica de protección de contraseña MQCSP de IBM MQ , si la conexión no utiliza el cifrado TLS.

Este atributo puede ser leído por clientes C, .NET no gestionados, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gestionados y XMS .NET gestionados.

La protección por contraseña MQCSP es útil para fines de prueba y desarrollo porque utilizar la protección por contraseña MQCSP es más sencillo que establecer el cifrado TLS, pero no es tan seguro.

Para obtener más información sobre la protección de credenciales en la estructura MQCSP y los valores que se pueden establecer para este atributo, consulte Protección de contraseña MQCSP.

Tareas relacionadas
Conexión de aplicaciones MQI de IBM MQ con gestores de colas

Multi Stanza ClientExitPath del archivo de configuración de cliente

Utilice la stanza ClientExitPath para especificar las ubicaciones predeterminadas de las salidas de canal en el cliente.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

Los siguientes atributos pueden incluirse en la stanza ClientExitPath:

ExitsDefaultPath = *cadena*
Especifica la ubicación de las salidas de canal de 32 bits para los clientes.

Este atributo es legible por clientes C, .NET no gestionados, .NET gestionados, XMS .NET gestionados, IBM MQ classes for Java y IBM MQ classes for JMS. Los clientes IBM MQ classes for Java y IBM MQ classes for JMS utilizan este atributo para localizar salidas de canal de 32-bits que no se escriben en Java.

ExitsDefaultPath64 = *cadena*
Especifica la ubicación de las salidas de canal de 64 bits para los clientes.

Este atributo es legible por clientes C, .NET no gestionados, .NET gestionados, XMS .NET gestionados, IBM MQ classes for Java y IBM MQ classes for JMS. Los clientes IBM MQ classes for Java y IBM MQ classes for JMS utilizan este atributo para localizar salidas de canal de 64-bits que no se escriben en Java.

JavaExitsClassPath = *cadena*
Los valores que deben añadirse a la vía de acceso de clases cuando se ejecuta una salida de Java. Estos los ignoran las salidas en cualquier otro idioma.

Este atributo puede ser leído por clientes IBM MQ classes for Java y IBM MQ classes for JMS.

En el archivo de configuración de JMS, el nombre JavaExitsClassPath recibe el prefijo estándar com.ibm.mq.cfg. y este nombre completo también se utiliza en la propiedad del sistema IBM MQ.

Multi Stanza de conexión del archivo de configuración de cliente

Utilice la stanza de conexión para especificar un nombre de aplicación.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

El atributo siguiente se puede incluir en la stanza de conexión:

ApplName = ExampleApplname
Puede especificar un nombre de aplicación en el archivo de configuración de cliente.

Este atributo puede ser utilizado por C y clientes de .NET no gestionados.

Multi Stanza JMQI del archivo de configuración de cliente

Utilice la stanza JMQI para especificar parámetros de configuración para la Java Message Queuing Interface (JMQI) utilizada por IBM MQ classes for Java y IBM MQ classes for JMS.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

El atributo siguiente se puede incluir en la stanza JMQI:

useMQCSPauthentication = NO | YES
Controla si las aplicaciones IBM MQ classes for Java y IBM MQ classes for JMS tiene que usar el modo de compatibilidad o el modo de autenticación MQCSP al autenticarse con un gestor de colas.

Este atributo puede ser leído por clientes IBM MQ classes for Java y IBM MQ classes for JMS.

Este atributo puede tener los valores siguientes:

NO
Utilice la modalidad de compatibilidad al autenticar con un gestor de colas. Este es el valor predeterminado en versiones anteriores a IBM MQ 9.3.0.

SÍ
Utilice la modalidad de autenticación MQCSP al autenticar con un gestor de colas. Es el valor predeterminado de IBM MQ 9.3.0.

Hay otras formas de establecer la modalidad de autenticación que tienen prioridad sobre el valor del atributo **useMQCSPauthentication**. Si desea más información sobre la modalidad de compatibilidad y la modalidad de autenticación MQCSP, consulte Autenticación de conexión con el cliente Java.

Windows Stanzas LU62, NETBIOS y SPX el archivo de configuración de cliente

Sólo en sistemas Windows, utilice estas stanzas para especificar parámetros de configuración para los protocolos de red especificados.

stanza LU62

utilice la stanza LU62 para especificar parámetros de configuración de protocolo SNA LU 6.2. Los siguientes atributos pueden incluirse en esta stanza:

Library1 = *NombreDLL*|WCPIC32
El nombre de la DLL de APPC.

Library2 = *NombreDLL*|WCPIC32
Igual que Library1, utilizada si el código se almacena en dos bibliotecas distintas.

Nombre TP
El nombre de TP que debe iniciarse en la ubicación remota.

stanza NETBIOS

Utilice la stanza NETBIOS para especificar parámetros de configuración de protocolo NetBIOS. Los siguientes atributos pueden incluirse en esta stanza:

AdapterNum = *número*|0
El número del adaptador de la LAN.

Library1 = *NombreDLL*|NETAPI32
El nombre de la DLL de NetBIOS.

LocalName = *nombre*
El nombre por el que este ordenador es conocido en la LAN.

Esto es equivalente a la variable de entorno MQNAME.

NumCmds = *número*|1
La cantidad de mandatos para asignar.

NumSess = *número*|1
La cantidad de sesiones para asignar.

stanza SPX

Utilice la stanza SPX para especificar parámetros de configuración de protocolo SPX. Los siguientes atributos pueden incluirse en esta stanza:

BoardNum = *número*|0
El número de adaptador de la LAN.

KeepAlive = YES|NO
Permite activar o desactivar la función KeepAlive.

KeepAlive=YES hace que SPX compruebe periódicamente que el otro extremo de la conexión siga disponible. En caso contrario se cierra el canal.

Library1 = *NombreDLL*|WSOCK32.DLL
Nombre DLL de SPX.

Library2 = *NombreDLL*|WSOCK32.DLL
Igual que Library1, utilizada si el código se almacena en dos bibliotecas distintas.

Socket = *número*|5E86
Número de socket de SPX en notación hexadecimal.

Multi Stanza MessageBuffer del archivo de configuración de cliente

Utilice la stanza MessageBuffer para especificar la información sobre almacenamiento intermedio de mensajes.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

Los siguientes atributos pueden incluirse en la stanza MessageBuffer:

MaximumSize = *entero*|1
Tamaño, en kilobytes, del almacenamiento intermedio de lectura anticipada, en el intervalo 1 a 999.999.

Este atributo puede ser leído por clientes C, .NET no gestionados, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gestionados y XMS .NET gestionados.

Existen los valores especiales siguientes:

-1
El cliente determina el valor adecuado.

0
La lectura anticipada está inhabilitada para el cliente.

PurgeTime = *entero*|600

Intervalo, en segundos, tras el cual se purgan los mensajes dejados en el almacenamiento intermedio de lectura anticipada.

Este atributo puede ser leído por clientes C, .NET no gestionados, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gestionados y XMS .NET gestionados.

Si la aplicación cliente está seleccionando mensajes basándose en el MsgId o CorrelId es posible que el almacenamiento intermedio de lectura anticipada pueda contener mensajes enviados al cliente con un MsgId o CorrelId solicitado anteriormente. Estos mensajes quedarán abandonados en el almacenamiento intermedio de lectura anticipada hasta que se emita una llamada MQGET con un MsgId o CorrelId adecuado. Puede depurar mensajes del almacenamiento intermedio de lectura anticipada estableciendo PurgeTime. Todos los mensajes que hayan permanecido en el almacenamiento intermedio de lectura anticipada durante un periodo de tiempo superior al intervalo de depuración se depurarán automáticamente. Estos mensajes ya se han eliminado de la cola en el gestor de colas, de modo que, a menos que se estén examinando, se pierden.

El intervalo válido se encuentra en el rango 1 a 999.999 segundos, o el valor especial 0, que significa que no se realiza la depuración.

UpdatePercentage = *entero*|-1

El valor del porcentaje de actualización, en el intervalo 1 - 100, utilizado para calcular el valor de umbral a determinar cuando una aplicación cliente realiza una solicitud nueva al servidor. El valor especial -1 indica que e cliente determina el valor apropiado.

Este atributo puede ser leído por clientes C, .NET no gestionados, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gestionados y XMS .NET gestionados.

El cliente envía periódicamente una solicitud al servidor indicando cuántos datos ha consumido la aplicación cliente. Se envía una solicitud cuando el número de bytes, *n*, recuperados por el cliente mediante llamadas MQGET excede un umbral *T*. *n* se restablece en cero cada vez que se envía una nueva solicitud al servidor.

El umbral T se calcula de la manera siguiente:

```
T = Upper - Lower
```

Upper es lo mismo que el tamaño de almacenamiento intermedio de lectura anticipada, especificado por el atributo *MaximumSize*, en Kilobytes. Su valor predeterminado es 100 Kb.

Lower es inferior a Upper y se especifica mediante el atributo *UpdatePercentage*. Este atributo es un número que se encuentra en el rango entre 1 y 100, y tiene un valor predeterminado de 20. Lower se calcula de la manera siguiente:

```
Lower = Upper x UpdatePercentage / 100
```

Ejemplo 1:

Los atributos MaximumSize y UpdatePercentage tiene los valores predeterminados de 100 Kb y 20 Kb.

El cliente llama a MQGET para recuperar un mensaje y lo hace de forma repetida. Esto continúa hasta que MQGET ha consumido n bytes.

Utilizando el cálculo

```
T = Upper - Lower
```

T es (100 - 20) = 80 Kb.

De forma que cuando las llamadas MQGET han eliminado 80 Kb de una cola, el cliente realiza una nueva solicitud automáticamente.

Ejemplo 2:

Los atributos MaximumSize tienen el valor predeterminado de 100 Kb y se elige para UpdatePercentage un valor de 40.

El cliente llama a MQGET para recuperar un mensaje y lo hace de forma repetida. Esto continúa hasta que MQGET ha consumido n bytes.

Utilizando el cálculo

```
T = Upper - Lower
```

T es (100 - 40) = 60 Kb

De forma que cuando las llamadas MQGET han eliminado 60 Kb de una cola, el cliente realiza una nueva solicitud automáticamente. Esto tiene lugar antes que en el EJEMPLO 1 donde se utilizaron valores predeterminados.

Por consiguiente, elegir un umbral *T* más grande tiende a disminuir la frecuencia a la que se envían las solicitudes del cliente al servidor. Y a la inversa, elegir un umbral *T* más pequeño tiende a incrementar la frecuencia a la que se envían las solicitudes del cliente al servidor.

No obstante, si elige un umbral *T* grande puede significar que se ha reducido la ganancia de rendimiento de lectura anticipada, ya que se puede incrementar la posibilidad de que el almacenamiento intermedio de lectura anticipada esté vacío. Cuando esto sucede, puede que una llamada MQGET haya tenido que detenerse mientras esperaba que llegaran datos del servidor.

Multi Stanza PreConnect del archivo de configuración del cliente

Utilice la stanza PreConnect para configurar la salida PreConnect en el archivo `mqclient.ini`.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

Los siguientes atributos se pueden incluir en la stanza PreConnect:

Data = *datos_usuario*

Este atributo especifica los datos de usuario que se pasan a la salida Preconnect. Los datos que se pasan a la salida Preconnect son específicos de la implementación de la salida Preconnect que se utiliza y de los datos que se espera que pase.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Por ejemplo, este atributo se podría utilizar para especificar el URL del repositorio donde se almacenan las definiciones de conexión, por ejemplo, cuando se utiliza un servidor LDAP:

```
Data = ldap://myLDAPServer.com:389/cn=wmq,ou=ibm,ou=com
```

Function = *miFunc*

Nombre del punto de entrada funcional a la biblioteca que contiene el código de salida de Preconnect.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

La definición de función se adhiere al prototipo de salida de Preconnect MQ_PRECONNECT_EXIT.

La longitud máxima de este campo es MQ_EXIT_NAME_LENGTH.

Module = *miMod*

El nombre del módulo que contiene el código de salida de la API.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Si este campo contiene el nombre de vía de acceso completo del módulo, se utiliza tal cual.

Sequence = *número_secuencia*

La secuencia en la que se llama a esta salida en relación con otras salidas. Se llama antes a una salida con un número de secuencia bajo que a una salida con un número de secuencia más alto. No es

necesario que los números de secuencia de las salidas sean continuos; una secuencia de 1, 2, 3 tiene el mismo resultado que una secuencia de 7, 42, 1096. Este atributo es un valor numérico sin signo.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Se pueden definir varias stanzas PreConnect dentro del archivo `mqclient.ini`. El orden de proceso de cada salida está determinado por el atributo Sequence de la stanza.

Tareas relacionadas

Referencia a las definiciones de conexión utilizando una salida de preconexión desde un depósito

Multi Stanza de seguridad del archivo de configuración de cliente

Utilice la stanza Security para inhabilitar o habilitar AMS para las conexiones de cliente a un gestor de colas.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

El atributo siguiente se puede incluir en la stanza Security:

DisableClientAMS = NO|YES

El atributo DisableClientAMS le permite inhabilitar IBM MQ Advanced Message Security (AMS) si está utilizando un cliente IBM MQ para conectarse a un gestor de colas desde una versión anterior del producto y se notifica un error `2085 (MQRC_UNKNOWN_OBJECT_NAME)`.

IBM MQ Advanced Message Security (AMS) se habilita automáticamente en un cliente IBM MQ y, por lo tanto, de forma predeterminada, el cliente intenta comprobar las políticas de seguridad para los objetos en el gestor de colas.

En los ejemplos siguientes se muestra cómo utilizar el atributo DisableClientAMS:

- Para inhabilitar AMS:

```
Security:
DisableClientAMS=Yes
```

- Para habilitar AMS:

```
Security:
DisableClientAMS=No
```

Los clientes C, IBM MQ classes for Java y IBM MQ classes for JMS pueden leer este atributo.

MQIInitialKeyArchivo = *nombre_vía_acceso*

La vía de acceso completa y el nombre del archivo que contiene la clave inicial que se ha utilizado para cifrar las credenciales proporcionadas por el cliente. Se debe especificar la clave inicial si se ha especificado un archivo de claves inicial cuando se cifró la frase de contraseña del repositorio de claves utilizando el programa de utilidad **runmqicred**.

Este atributo puede ser leído por C y los clientes .NET no gestionados.

Tareas relacionadas

Inhabilitación de Advanced Message Security en el cliente

Multi Stanza SSL del archivo de configuración de cliente

Utilice la stanza SSL para especificar información sobre la utilización de TLS.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

Los siguientes atributos pueden incluirse en la stanza SSL:

OutboundSNI = CHANNEL | HOSTNAME

Si **OutboundSNI** se establece en CANAL, los clientes con capacidad SNI establecen SNI en el nombre de canal de IBM MQ de destino en el sistema remoto al iniciar una conexión TLS.

Si este atributo se establece en HOSTNAME, los clientes que tengan habilitado SNI establecerán la cabecera SNI en el nombre de host, lo que hará que las solicitudes de conexión saliente reciban el certificado predeterminado del gestor de colas remoto durante el reconocimiento TLS, por lo que no se podrán utilizar los certificados por canal.

Este atributo puede ser leído por clientes C, .NET, IBM MQ classes for Java y IBM MQ classes for JMS.

El cliente Java/JMS interpreta los valores de propiedad con distinción de mayúsculas y minúsculas, por lo que los valores YES/NO deben establecerse en mayúsculas.

A partir de IBM MQ 9.3.0, el cliente IBM MQ gestionado .NET se ha actualizado para establecer SERVERNAME en el nombre de host respectivo si la propiedad **OutboundSNI** se establece en HOSTNAME, lo que permite a un cliente IBM MQ gestionado .NET conectarse a un gestor de colas utilizando Red Hat OpenShift routes.

Nota: Si una aplicación con un valor **OutboundSNI** de HOSTNAME se conecta a un canal con una etiqueta de certificado configurada, la aplicación se rechaza con un MQRC_SSL_INITIALIZATION_ERROR y se imprime un mensaje AMQ9673 en los registros de errores del gestor de colas.

AllowOutboundSNI = YES (valor predeterminado) | NO

Si está habilitado, los clientes que tengan habilitado SNI establecerán SNI en el nombre de canal IBM MQ de destino en el sistema remoto al iniciar una conexión TLS. Si este atributo se establece en NO, los clientes que tengan habilitado SNI no establecerán la cabecera SNI lo que hará que las solicitudes de conexión saliente reciban el certificado predeterminado del gestor de colas remoto durante el reconocimiento TLS, por lo que no se podrán utilizar los certificados por canal.

Este atributo puede ser leído por clientes C, .NET, IBM MQ classes for Java y IBM MQ classes for JMS.

El cliente Java/JMS interpreta los valores de propiedad con distinción de mayúsculas y minúsculas, por lo que los valores YES/NO deben establecerse en mayúsculas.

Atención: Deprecated En IBM MQ 9.3.0 la propiedad **AllowOutboundSNI** está en desuso y solo está disponible para fines de compatibilidad con versiones anteriores.

AllowOutboundSNI establecido en YES proporciona la misma función que **OutboundSNI** establecida en CHANNEL, mientras que **AllowOutboundSNI** establecido en NO proporciona la misma función que **OutboundSNI** establecido en HOSTNAME.

Si los atributos **AllowOutboundSNI** y **OutboundSNI** están presentes en la stanza SSL, el valor de **OutboundSNI** tiene prioridad.

IBM i ALW **AllowTLSV13 =Y | YES | T | TRUE (valor predeterminado) | N | NO | F | FALSE**

Especifica si un gestor de colas va a poder utilizar las CipherSpecs de TLS 1.3 (consulte Habilitación de CipherSpecs).

Este atributo puede ser leído por los clientes C/C++.

Este atributo tiene los siguientes valores posibles:

- Y (valor predeterminado), YES (valor predeterminado), T (valor predeterminado) o TRUE (valor predeterminado): habilita TLS 1.3 , lo que permite al gestor de colas utilizar las CipherSpecsde TLS 1.3 .
- N, NO, F o FALSE: inhabilita TLS 1.3, lo que significa que el gestor de colas no puede utilizar las CipherSpecs de TLS 1.3.

Nota: Cuando se utiliza el cliente MQI, el valor de **AllowTLSV13** se infiere a menos que se especifique explícitamente en la stanza SSL del archivo “Stanza SSL del archivo de configuración de cliente” en la página 199 que utiliza la aplicación. Para obtener más información, consulte Cliente de IBM MQ MQI y TLS 1.3.

CDPCheckExtensions = YES|NO (valor predeterminado)

CDPCheckExtensions especifica si los canales TLS de este gestor de colas intentan comprobar los servidores CDP especificados en las extensiones de certificado CrlDistributionPoint.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Este atributo tiene los siguientes valores posibles:

- YES (valor predeterminado): los canales TLS intentan comprobar los servidores CDP para determinar si se revoca un certificado digital.
- NO: los canales TLS no intentan comprobar los servidores CDP. Este valor es el valor por omisión.

CertificateLabel = *cadena*

la etiqueta de certificado de la definición de canal.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Consulte Etiqueta de certificado (CERTLABL) para obtener más información.

CertificateValPolicy = *cadena*

Determina el tipo de validación de certificados utilizado.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Este atributo tiene los siguientes valores posibles:

CUALQUIERA
Utilizar cualquier política de validación de certificados soportada por la biblioteca de sockets seguros subyacente. Este valor es el predeterminado.

RFC5280
Utilizar sólo la validación de certificados que cumpla con el estándar RFC 5280.

V 9.4.0 V 9.4.0 **NINGUNO**
No utilizar validación de certificado.

ClientRevocationChecks = REQUIRED|OPTIONAL|DISABLED

Determina cómo se configura la comprobación de revocación de certificados si la llamada de conexión del cliente utiliza un canal TLS. Consulte también **OCSPAuthentication**.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Este atributo tiene los siguientes valores posibles:

REQUIRED (valor predeterminado)
Intenta cargar la configuración de revocación de certificados desde CCDT y realiza la comprobación de revocación de certificados, tal como se ha configurado. Si el archivo CCDT no se puede abrir o si no es posible validar el certificado (por ejemplo, debido a que no está disponible un servidor OCSP o CRL), llamada MQCONN fallará. No se realiza la comprobación de la revocación si CCDT no contiene ninguna configuración de revocación pero esto no hace que el canal falle.

Windows En los sistemas Windows, también puede utilizar Active Directory para la comprobación de revocación de CRL. No puede utilizar Active Directory para la comprobación de revocación de OCSP.

Si está utilizando MQSCO o CCDT, la conexión se realiza correctamente. Si no hay ningún archivo CCDT y si tampoco se proporciona MQSCO, la conexión falla con un código de razón 2059 y el registro de errores informa de `AMQ9518E: Archivo '/var/mqm/AMQCLCHL.TAB' no encontrado`.

Opcional
Igual que para REQUIRED, pero si no se puede cargar la configuración de revocación de certificado, el canal no fallará.

DISABLED
No se intenta cargar la configuración de revocación de certificados desde CCDT y no se realiza la comprobación de revocación de certificados.

Nota: Si está utilizando MQCONNX, en lugar de las llamadas MQCONN, puede optar por proporcionar los registros de información de autenticación (MQAIR) a través de MQSCO. Por lo tanto, el comportamiento predeterminado de MQCONNX es que no dará error si no se puede abrir el archivo CCDT pero asumirá que está suministrando un MQAIR (incluso si ha elegido no hacerlo).

EncryptionPolicySuiteB = *cadena*
Determina si un canal utiliza cifrado compatible con Suite B y qué nivel de potencia se utilizará.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Este atributo tiene los siguientes valores posibles:

NINGUNO
No se utiliza el cifrado compatible con Suite B. Este valor es el predeterminado.

128_BIT,192_BIT
Establece el nivel de seguridad para niveles de 128 bit y de 192 bits.

128_BIT
Establece la potencia de seguridad en un nivel de 128 bits.

192_BIT
Establece la potencia de seguridad en un nivel de 192 bits.

ALW **EnvironmentScope=PROCESS|CONNECTION**
Controla si IBM MQ utiliza un único entorno IBM Global Security Kit (GSKit) para todo el proceso o un entorno GSKit por conexión.

Este atributo puede ser leído por clientes C.

Este atributo tiene los siguientes valores posibles:

PROCESS
Se utiliza un único entorno de GSKit para varias conexiones creadas por el proceso. La utilización de este valor significa que los cambios del almacén de claves TLS no estarán disponibles hasta que se hayan detenido todas las conexiones TLS activas dentro del proceso.

Este es el valor predeterminado.

CONNECTION
Se crea un entorno GSKit para cada conexión dentro del mismo proceso. Habilitarlo significa que los cambios de almacén de claves TLS se recogerán inmediatamente mediante cualquier nueva conexión TLS que inicie el proceso.

Aviso: La habilitación de esta modalidad de operación hace que las aplicaciones utilicen recursos adicionales de CPU y memoria para crear cada entorno de GSKit . Este consumo de recursos aumenta con cada conexión TLS simultánea adicional.

ALW **MinimumRSAKeySize=int**

Especifica el tamaño de clave mínimo que deben tener los certificados RSA para poder aceptarse. Permite cualquier valor igual a 0 o superior. Toma de forma predeterminada el valor 1, si no se especifica.

Este atributo puede ser leído por los clientes C/C++.

OCSPAuthentication = OPTIONAL|REQUIRED|WARN
Define el comportamiento de IBM MQ cuando se habilita OCSP y la comprobación de revocación de OCSP no puede determinar el estado de revocación del certificado. Consulte también **ClientRevocationChecks**.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Este atributo tiene los siguientes valores posibles:

Opcional
Se acepta cualquier certificado que tenga un estado de revocación que no se pueda determinar mediante la comprobación de OCSP, y no se genera ningún mensaje de aviso o de error. La conexión SSL o TLS continúa como si no se hubiera realizado ninguna comprobación de revocación.

OBLIGATORIO
La comprobación de OCSP debe producir un resultado de revocación definitivo para cada certificado SSL o TLS que se haya comprobado. Cualquier certificado SSL o TLS que tenga un estado de revocación que no se pueda comprobar se rechaza, y se emite un mensaje de error. Si se habilitan mensajes de sucesos SSL del gestor de colas, se genera un mensaje MQRC_CHANNEL_SSL_ERROR con un ReasonQualifier de MQRQ_SSL_HANDSHAKE_ERROR. Se cierra la conexión.

Este es el valor predeterminado.

WARN
Si una comprobación de revocación OCSP no puede determinar el estado de revocación de cualquier certificado SSL o TLS, se informa de un error en los registros de errores del gestor de colas. Si se habilitan los mensajes de sucesos SSL del gestor de colas, se genera un mensaje MQRC_CHANNEL_SSL_WARNING con un ReasonQualifier de MQRQ_SSL_UNKNOWN_REVOCATION. La conexión tiene permiso para continuar.

OCSPCheckExtensions=YES|NO
Controla si IBM MQ actúa en las extensiones de certificado AuthorityInfoAccess.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Si el valor se establece en NO, IBM MQ ignora las extensiones de certificado AuthorityInfoAccess, y no intenta efectuar una comprobación de seguridad de OCSP. El valor predeterminado es YES.

ALW **OCSPTimeout = *número***
El número de segundos que se debe esperar un programa de respuesta OCSP al realizar una comprobación de revocación.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

A partir de IBM MQ 9.3.0, si se establece un valor de 0, se utiliza el tiempo de espera predeterminado de 30 segundos.

Si no se ha establecido ningún valor, se utiliza el valor predeterminado de IBM MQ de 30 segundos.

ALW **PeerCertChainValidation=*serie***
Este atributo puede ser leído por C y los clientes .NET no gestionados.

La serie puede ser uno de los dos valores siguientes:

- `Usepeerchain` **[Valor predeterminado]**: la cadena de certificados proporcionada por el interlocutor se puede utilizar de puente para cualquier intervalo de cadena de confianza al validar los certificados. Con la excepción del certificado raíz.
- `TruststoreOnly` **[No recomendado]**: solo se utilizarán certificados del almacén de confianza para validar el certificado del interlocutor.

SSLCryptoHardware = *cadena*
Establece la serie de parámetros necesaria para configurar el hardware de cifrado PKCS #11 existente en el sistema.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Especifique una serie con el formato siguiente: `GSK_PKCS11 = driver path and filename;token label;token password;symmetric cipher setting;`

Por ejemplo: `GSK_PKCS11=/usr/lib/pkcs11/PKCS11_API.so;tokenlabel;passw0rd;SYMMETRIC_CIPHER_ON`

La vía de acceso del controlador es una vía de acceso absoluta a la biblioteca compartida que ofrece soporte para la tarjeta PKCS #11. El nombre de archivo de controlador es el nombre de la biblioteca compartida. Un ejemplo del valor necesario para la vía de acceso del controlador PKCS #11 y el nombre de archivo es `/usr/lib/pkcs11/PKCS11_API.so`. Para acceder a las operaciones de cifrado simétrico a través de GSKit, especifique el parámetro de valor de cifrado simétrico. El valor de este parámetro es:

SYMMETRIC_CIPHER_OFF
No acceder a operaciones de cifrado simétrico. Este valor es el predeterminado.

SYMMETRIC_CIPHER_ON
Acceder a las operaciones de cifrado simétrico.

Linux AIX Al suministrar los diferentes componentes de la serie, debe escapar los caracteres de punto y coma utilizando el carácter de barra inclinada invertida, ya que el carácter de punto y coma se trata como un comentario. Por ejemplo: '`\;`'

Debe proteger la contraseña de señal contenida en la serie del atributo **SSLCryptoHardware**. Para obtener más información, consulte Clientes deIBM MQ que utilizan hardware de cifrado .

Para manejar las contraseñas cifradas, ahora no hay límite para la longitud de la serie.

El valor predeterminado es en blanco. Si especifica una serie que no está en el formato correcto, se genera un error.

SSLCryptoHardwareKeyFile = *nombre_vía_acceso*
La vía de acceso completa y el nombre del archivo que contiene la clave inicial que se ha utilizado para cifrar la contraseña en la serie de configuración de hardware criptográfico PKCS #11 que se especifica con el atributo **SSLCryptoHardware** . La clave inicial debe especificarse si se ha especificado un archivo de claves inicial cuando la contraseña de la serie de configuración de hardware criptográfico se cifró utilizando el mandato **runp11cred** . Para obtener más información, consulte Clientes de IBM MQ que utilizan hardware criptográfico.

Este atributo puede ser leído por C y los clientes .NET no gestionados.

SSLFipsRequired = YES|NO
Especifica si sólo se van a utilizar algoritmos certificados por FIPS si se lleva a cabo el cifrado en IBM MQ.

Este atributo puede ser leído por C y los clientes .NET no gestionados.

Si se ha configurado el hardware de cifrado, los módulos criptográficos utilizados son aquellos módulos proporcionados por el producto de hardware. Estos pueden estar o no certificados por FIPS en un nivel determinado, dependiendo del producto de hardware que se esté utilizando.

SSLHTTPProxyName = *cadena*
La serie es el nombre de host o la dirección de red del servidor proxy HTTP que GSKit debe utilizar para las comprobaciones de OCSP. Esta dirección puede ir seguida de un número de puerto opcional, delimitado mediante paréntesis. Si no especifica el número de puerto, se utiliza el puerto HTTP predeterminado, el 80.

Este atributo puede ser leído por C y los clientes .NET no gestionados.

AIX Para los clientes de de 32 bits en AIX, la dirección de red sólo puede ser una dirección IPv4.

En otras plataformas, la dirección de red puede ser una dirección IPv4 o IPv6.

Este atributo puede ser necesario si, por ejemplo, un cortafuegos impide el acceso al URL del programa de respuestas OCSP.

ALW **SSLHTTPConnectTimeout = *número*|0**
El número de segundos que se va a esperar a que una conexión de red se establezca correctamente en un servidor HTTP al realizar una comprobación de revocación.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

Si no se establece ningún valor, se utiliza el valor predeterminado de IBM MQ de 0 (desactivado).

SSLKeyRepository = *nombreruta*
La vía de acceso completa y el nombre de archivo del repositorio de claves que contiene el certificado digital del usuario. Si no se especifica la extensión de archivo, se presupone que es `.kdb`.

Este atributo puede ser leído por C y los clientes .NET no gestionados.

SSLKeyRepositoryPassword = *frase de contraseña*
Frase de contraseña para acceder al repositorio de claves. El valor puede ser una serie de texto sin formato o una frase de contraseña que se ha cifrado utilizando el programa de utilidad **`runmqicred`** .

Este atributo puede ser leído por C y los clientes .NET no gestionados.

SSLKeyResetCount = *entero*|`0`
El número de bytes no cifrados enviados y recibidos en un canal TLS antes de que se cambie la clave secreta.

Este atributo puede ser leído por C y los clientes .NET no gestionados.

El valor debe estar entre 0 y 999999999.

El valor predeterminado es 0, lo que significa que las claves secretas no se negocian nunca.

Si especifica un valor entre 1 y 32768, los canales TLS utilizan un número de restablecimiento de clave secreta de 32768 (32 Kb). De esta forma, se evitan restablecimientos de clave excesivos que se producirían para valores de restablecimiento de claves secretas pequeñas.

Multi Stanza TCP del archivo de configuración de cliente

Utilice la stanza TCP para especificar los parámetros de configuración de protocolo de red TCP.

Nota: La descripción de cada atributo de esta sección indica qué clientes de IBM MQ pueden leer este atributo. Para ver una tabla de resumen de todas las secciones del archivo de configuración de IBM MQ MQI client, consulte Qué atributos de IBM MQ puede leer cada cliente.

Los siguientes atributos pueden incluirse en la stanza TCP:

ClntRcvBuffSize = *número*|`0`
El tamaño en bytes del almacenamiento intermedio de recepción TCP/IP utilizado por el extremo del cliente de un canal de conexión de cliente - conexión de servidor.

Este atributo puede ser leído por clientes C, .NET no gestionados, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gestionados y XMS .NET gestionados.

Un valor de cero indica que el sistema operativo gestionará los tamaños de almacenamiento intermedio, en lugar de que IBM MQ fije los tamaños de almacenamiento intermedio. Si el valor se establece en cero, se utilizan los valores predeterminados de sistema operativo. Si no se establece ningún valor, se utiliza el valor predeterminado de IBM MQ, 32768.

ClntSndBuffSize = *número*|`0`
El tamaño en bytes del almacenamiento intermedio de envío TCP/IP utilizado por el extremo del cliente del canal de conexión del cliente y del servidor.

Este atributo puede ser leído por clientes C, .NET no gestionados, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gestionados y XMS .NET gestionados.

Un valor de cero indica que el sistema operativo gestionará los tamaños de almacenamiento intermedio, en lugar de que IBM MQ fije los tamaños de almacenamiento intermedio. Si el valor se establece en cero, se utilizan los valores predeterminados de sistema operativo. Si no se establece ningún valor, se utiliza el valor predeterminado de IBM MQ, 32768.

Connect_Timeout = *número*
El número de segundos antes de que un intento de conectar el socket sobrepase el tiempo de espera.

Si **ConnectTimeout** = 0, y se emite SOCK_NONBLOCK antes de una llamada a connect () asíncrona, la llamada no está bloqueada. El valor de tiempo de espera predeterminado de 20 segundos (CONNECT_WAIT_MAX) es aplicable para comprobar el estado de conexión.

Este atributo puede ser leído por clientes C, .NET, IBM MQ classes for Java y IBM MQ classes for JMS.

Los procesos de canal de IBM MQ se conectan a través de sockets no de bloqueo. Por lo tanto, si el otro extremo del socket no está preparado, connect() se devuelve inmediatamente con *EINPROGRESS* o *EWOULDBLOCK*. Después de esto, no hay ningún intento de volver a conectarse.

Si Connect_Timeout se establece en un valor distinto de cero, IBM MQ espera el período estipulado durante la llamada select() a que el socket esté preparado. Esto aumenta las posibilidades de éxito de una llamada connect() posterior. Esta opción podría ser conveniente en situaciones en las que las conexiones requerirían algún período de espera, debido a una gran carga en la red.

No hay relación entre los parámetros Connect_Timeout, ClntSndBuffSize y ClntRcvBuffSize.

IPAddressVersion = MQIPADDR_IPV4|MQIPADDR_IPV6
Especifica el protocolo IP que se tiene que utilizar en una conexión de canal.

Este atributo puede ser leído por clientes C, .NET no gestionados, .NET gestionados y XMS .NET gestionados.

Tiene los posibles valores de serie MQIPADDR_IPV4 o MQIPADDR_IPV6. Estos valores tienen los mismos significados que IPV4 y IPV6 en **ALTER QMGR IPADDRV** y la variable de entorno **MQIPADDRV** .

KeepAlive = YES|NO
Permite activar o desactivar la función KeepAlive. KeepAlive=YES hace que TCP/IP compruebe periódicamente si el otro extremo de la conexión sigue estando disponible. En caso contrario se cierra el canal.

Este atributo puede ser leído por clientes C, .NET no gestionados, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gestionados y XMS .NET gestionados.

Windows **Library1 = *NombreDLL*|WSOCK32**
(Sólo Windows) El nombre de la DLL de sockets TCP/IP.

Este atributo puede ser leído por clientes C y no gestionados de .NET.

V 9.4.0 Stanza de rastreo del archivo de configuración de cliente

Utilice la stanza Trace para habilitar el rastreo para las bibliotecas de cliente IBM MQ .NET y XMS .NET .

Los atributos siguientes se pueden incluir en la stanza TRACE:

MQDotnetTraceLevel=0 (valor predeterminado) |1|2
El atributo **MQDotnetTraceLevel** se utiliza para iniciar o detener el rastreo de IBM MQ .NET :

- 0: Detiene el rastreo: este es el valor predeterminado.
- 1: Inicia el rastreo con menos detalles.
- 2: Inicia el rastreo con los detalles completos, recomendados.

Este atributo lo puede leer el cliente IBM MQ .NET gestionado.

MQDotnetTracePath =*nombre_vía_acceso*
El atributo **MQDotnetTracePath** apunta a una carpeta donde se crearán los archivos de rastreo de IBM MQ .NET . El directorio actual de la aplicación se utiliza si la vía de acceso está en blanco o la propiedad **MQDotnetTracePath** no está definida.

Este atributo lo puede leer el cliente IBM MQ .NET gestionado.

MQDotnetErrorPath =*nombre_vía_acceso*
El atributo **MQDotnetErrorPath** apunta a una carpeta en la que se crearán archivos de registro de errores para el rastreo de IBM MQ .NET . El directorio actual de la aplicación se utiliza si la vía de acceso está en blanco o si el atributo **MQDonetErrorPath** no está definido.

Este atributo lo puede leer el cliente IBM MQ .NET gestionado.

XMSDotnetTraceLevel=0 (valor predeterminado) |1|2
El atributo **XMSDotnetTraceLevel** se utiliza para iniciar o detener el rastreo de XMS .NET :

- 0: Detiene el rastreo: este es el valor predeterminado.
- 1: Inicia el rastreo con el formato básico.
- 2: Inicia el rastreo con formato avanzado.

El cliente XMS .NET gestionado puede leer este atributo.

XMSDotnetTraceFilePath=*nombre_archivo*
Si no se establece un valor para el atributo **XMSDotnetTraceFilePath** , o si este atributo está presente pero contiene una serie vacía, el archivo de rastreo para XMS .NET se coloca en el directorio actual. Para almacenar el archivo de rastreo en un directorio con nombre, especifique el nombre de directorio en **XMSDotnettraceFilePath**, por ejemplo, `XMSDotnetTraceFilePath="c:\somepath"`.

El cliente XMS .NET gestionado puede leer este atributo.

XMSDotnetTraceSpecification =*ComponentName=type=state*
El atributo **XMSDotnetTraceSpecification** especifica el nombre de la clase que desea rastrear y el tipo de rastreo que necesita para XMS .NET:

- *Nombre_componente* es el nombre de la clase que desea rastrear. Puede utilizar un carácter comodín * en este nombre. Por ejemplo, `*=all=enabled` especifica que desea rastrear todas las clases y `IBM.XMS.impl.*=all=enabled` especifica que solo requiere el rastreo de API.
- *type* puede ser cualquiera de los siguientes tipos de rastreo: all, debug, event, EntryExit.
- *estado* puede ser habilitado o inhabilitado.

Puede unir en una serie varios elementos de rastreo utilizando un delimitador ':' (dos puntos).

El cliente XMS .NET gestionado puede leer este atributo.

XMSDotnetTraceFileSize=*tamaño*
El atributo **XMSDotnetTraceFileSize** especifica el tamaño máximo del archivo de rastreo que se debe generar para XMS .NET. El máximo predeterminado es 20 MB, que se especifica como `XMSDotnetTraceFileSize=20`.

El cliente XMS .NET gestionado puede leer este atributo.

XMSDotnetTraceFileNumber=*número*
El atributo **XMSDotnetTraceFileNumber** especifica el número de archivos de rastreo que se van a retener para XMS .NET. El valor predeterminado es 4 (un archivo activo y tres archivos de archivado). El número mínimo permitido es 2.

El cliente XMS .NET gestionado puede leer este atributo.

Tareas relacionadas
Rastreo de aplicaciones .NET de IBM MQ con mqclient.ini
Rastreo de aplicaciones XMS .NET con mqclient.ini

Multi Archivo de configuración de rastreo de actividad, mqat.ini

El archivo de configuración de rastreo de actividad, `mqat.ini`, se utiliza para configurar el comportamiento de rastreo de actividad. Este archivo se utiliza para definir el nivel y la frecuencia de notificación de datos de rastreo de actividad. El archivo también proporciona una forma de definir reglas para habilitar e inhabilitar el rastreo de actividad basándose en el nombre de una aplicación.

El archivo `mqat.ini` sigue el mismo formato de par de clave y parámetro-valor de stanza que los archivos `mqs.ini` y `qm.ini` . El archivo consta de una sola stanza, AllActivityTrace, que se utiliza para configurar el nivel y la frecuencia de los datos de rastreo de actividad de informes de forma predeterminada para todo el rastreo de actividad. El archivo también puede contener varias stanzas ApplicationTrace . Cada una de estas stanzas define una regla para el comportamiento de rastreo para una o más conexiones, basándose en la coincidencia del nombre de aplicación de las conexiones con la regla. Para obtener más información, consulte Rastreo de actividad de aplicación y Configuración del comportamiento de rastreo de actividad utilizando `mqat.ini`.

El gestor de colas aplica una serie de reglas para determinar qué valores de stanzas se deben utilizar para una conexión. Opcionalmente, puede alterar temporalmente los valores de frecuencia y nivel de rastreo global bajo la stanza de rastreo AllActivitypara aquellas conexiones que coincidan con una stanza ApplicationTrace . Para obtener más información, consulte Configuración del comportamiento de rastreo de actividad utilizando `mqat.ini`.

Ubicaciones de directorio

En los sistemas AIX and Linux y IBM i , `mqat.ini` se encuentra en el directorio de datos del gestor de colas, que es la misma ubicación que el archivo `qm.ini` .

En sistemas Windows, `mqat.ini` se encuentra en el directorio de datos del gestor de colas de `C:\Program Files\IBM\WebSphere MQ\qmgrs\queue_manager_name`. Los usuarios que ejecutan aplicaciones que hay que rastrear necesitan permiso para leer este archivo.

AllActivityStanza de rastreo del archivo mqat.ini

La stanza AllActivityTrace del archivo de configuración `mqat.ini` especifica los parámetros que se utilizan para configurar los niveles de rastreo para un gestor de colas.

Una única stanza de rastreo AllActivitydefine valores para el rastreo de actividad que se aplica a todas las conexiones de IBM MQ , a menos que se altere temporalmente.

Los valores individuales de la stanza AllActivityTrace se pueden alterar temporalmente mediante información más específica en una stanza ApplicationTrace.

Si se especifica más de una stanza AllActivityTrace, se utilizarán los valores de la última stanza. Los parámetros que faltan en la stanza AllActivityTrace seleccionada tomarán los valores predeterminados. Los parámetros y valores de las stanzas de rastreo AllActivityanteriores se ignoran.

ActivityInterval
: Intervalo de tiempo en segundos entre mensajes de rastreo. El rastreo de actividad no utiliza una hebra de temporizador, de modo que el mensaje de rastreo no se graba en el instante exacto en que el tiempo transcurre, se graba cuando se ejecuta la primera operación de MQI después de que haya transcurrido el intervalo de tiempo. Si este valor es 0, el mensaje de rastreo se graba cuando la conexión se desconecta (o cuando se alcanza el valor especificado en el parámetro ActivityCount). Toma de forma predeterminada 1.

ActivityCount
: Número de operaciones MQI entre mensajes de rastreo. Si este valor es 0, el mensaje de rastreo se graba cuando la conexión se desconecta (o cuando ha transcurrido el intervalo de actividad). Toma de forma predeterminada 100.

TraceLevel
: La cantidad de detalles de parámetro que se rastrea para cada operación. La descripción de las operaciones individuales detalla qué parámetros se incluyen para cada nivel de rastreo. Establézcalo en `LOW`, `MEDIUM`o `HIGH`. El valor predeterminado es `MEDIUM`.

TraceMessageData
: Cantidad de datos de mensaje que se rastrean en bytes para las operaciones MQGET, MQPUT, MQPUT1y Callback. Toma de forma predeterminada 0.

StopOnGetTraceMsg
Se puede establecer en ON u OFF. El valor predeterminado es ON.

SubscriptionDelivery
Se puede establecer en BATCHED o IMMEDIATE. Determina si los parámetros **ActivityInterval** y **ActivityCount** se van a utilizar cuando estén presentes una o más suscripciones de rastreo de actividad. Si se establece este parámetro en IMMEDIATE , los valores **ActivityInterval** y **ActivityCount** se alteran temporalmente con valores efectivos de 1 cuando los datos de rastreo tienen una suscripción coincidente. Cada registro de rastreo de actividad no se procesa por lotes con otros registros de la misma conexión y en lugar de ello se entrega a la suscripción inmediatamente sin retardo. El valor IMMEDIATE aumenta la sobrecarga de rendimiento de la recopilación de datos de rastreo de actividad. El valor predeterminado es BATCHED.

Tareas relacionadas
Configuración del comportamiento de rastreo de actividad utilizando mqat.ini

Multi Stanza ApplicationTrace del archivo mqat.ini

El archivo de configuración `mqat.ini` puede contener varias stanzas ApplicationTrace . Cada una de estas stanzas define una regla para el comportamiento de rastreo para una o más conexiones, basándose en la coincidencia del nombre de aplicación de las conexiones con la regla.

Puede establecer los valores siguientes para la stanza ApplicationTrace :

Rastreo
Conmutador de rastreo de actividad que se puede establecer en ON u OFF. El parámetro **Trace** es un parámetro obligatorio sin ningún valor predeterminado. Se puede utilizar en la stanza específica de la aplicación para determinar si el rastreo de actividad está activo para el ámbito de la stanza de aplicación actual. Tenga en cuenta que este valor altera temporalmente los valores **ACTVTRC** y **ACTVCONO** para el gestor de colas.

ApplName
El parámetro **ApplName** se especifica como una serie de caracteres y es un parámetro necesario sin ningún valor predeterminado. Este valor se utiliza para determinar a qué aplicaciones se aplica la stanza ApplicationTrace. Se compara con el valor **ApplName** de la estructura de contexto de salida de API (que es equivalente al MQMD de MQMD.PutApplName). El contenido del valor **ApplName** varía en función del entorno de aplicación.

En Multiplatforms, sólo la parte del nombre de archivo de MQAXC.ApplName se compara con el valor de la stanza. Los caracteres a la izquierda del separador de vía de acceso situado en el extremo derecho se ignoran cuando se efectúa la comparación.

Se puede utilizar un único carácter comodín (*) al final del valor de **ApplName** para que coincida con cualquier número de caracteres después de ese punto. Si el valor **ApplName** se establece en un único carácter comodín (*), el valor **ApplName** coincide con todas las aplicaciones.

IBM i ApplFunction
El parámetro **ApplFunction** se especifica como una serie de caracteres. El valor predeterminado es *. El valor de este parámetro se utiliza para calificar a qué programas de aplicación se aplica la stanza ApplicationTrace y el valor **ApplName** .

La stanza es opcional y sólo es válida para los gestores de colas de IBM i . Se puede utilizar un único carácter comodín (*) al final del valor de **ApplName** para que coincida con cualquier número de caracteres. Por ejemplo, una stanza ApplicationTrace que especifica **ApplName** = *** y **ApplFunction** = *AMQSPUT0* se aplica a todas las invocaciones del programa AMQSPUT0 desde cualquier trabajo.

ApplClass
El parámetro **ApplClass** define la clase de una aplicación y se puede establecer en los valores siguientes:

- USER
- MCA

- ALL (este es el valor predeterminado)

  Para obtener una explicación de cómo los valores de **AppType** se corresponden con las conexiones de IBM MQ , consulte la Tabla 3 en Configuración del comportamiento de rastreo de actividad utilizando mqat.ini.

Opcionalmente, los valores de frecuencia y nivel de rastreo global bajo la stanza de rastreo AllActivityse pueden alterar temporalmente para aquellas conexiones que coincidan con una stanza ApplicationTrace .

Los parámetros siguientes se pueden establecer en una stanza ApplicationTrace . Si no están establecidos, el valor se hereda de los valores de la sección de rastreo AllActivity :

ActivityInterval
Intervalo de tiempo en segundos entre mensajes de rastreo. El rastreo de actividad no utiliza una hebra de temporizador, de modo que el mensaje de rastreo no se graba en el instante exacto en que el tiempo transcurre, se graba cuando se ejecuta la primera operación de MQI después de que haya transcurrido el intervalo de tiempo. Si este valor es 0, el mensaje de rastreo se graba cuando la conexión se desconecta (o cuando se alcanza el valor especificado en el parámetro ActivityCount). Toma de forma predeterminada 1.

ActivityCount
Número de operaciones MQI entre mensajes de rastreo. Si este valor es 0, el mensaje de rastreo se graba cuando la conexión se desconecta (o cuando ha transcurrido el intervalo de actividad). Toma de forma predeterminada 100.

TraceLevel
La cantidad de detalles de parámetro que se rastrea para cada operación. La descripción de las operaciones individuales detalla qué parámetros se incluyen para cada nivel de rastreo. Establézcalo en `LOW`, `MEDIUM`o `HIGH`. El valor predeterminado es `MEDIUM`.

TraceMessageData
Cantidad de datos de mensaje que se rastrean en bytes para las operaciones MQGET, MQPUT, MQPUT1y Callback. Toma de forma predeterminada 0.

StopOnGetTraceMsg
Se puede establecer en `ON` u `OFF`. El valor predeterminado es `ON`.

Tareas relacionadas

Configuración del comportamiento de rastreo de actividad utilizando mqat.ini

Configuración de la gestión de colas distribuidas

En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

Antes de empezar

Antes de leer esta sección es útil tener una comprensión de los canales, colas y otros conceptos introducidos en Gestión de colas distribuidas y clústeres.

Si tiene que conectar dos gestores de colas que están en redes físicas distintas, o debe comunicarse a través de una cortafuegos, el uso de IBM MQ Internet Pass-Thru podría simplificar la configuración. Para obtener más información, consulte IBM MQ Internet Pass-Thru.

Procedimiento

- Utilice la información de los subtemas siguientes para conectar sus aplicaciones utilizando colas distribuidas:
  - “Técnicas de gestión de colas distribuidas de IBM MQ” en la página 211
  - “Introducción a la gestión de colas distribuidas” en la página 231
  - “Cómo enviar un mensaje a otro gestor de colas” en la página 234

- “Desencadenamiento de canales” en la página 255
- “Seguridad de mensajes” en la página 253
- ALW “Supervisión y control de canales en AIX, Linux, and Windows” en la página 263
- IBM i “Supervisión y control de canales en IBM i” en la página 287

Conceptos relacionados

“Setting up IBM MQ for z/OS” en la página 901
Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

Tareas relacionadas

“Configuración de conexiones entre el cliente y el servidor” en la página 16
Para configurar los enlaces de comunicación entre IBM MQ MQI clients y servidores, decida el protocolo de comunicación, defina las conexiones en ambos extremos del enlace, inicie un escucha y defina canales.

“Configuración de un clúster de gestores de colas” en la página 309
Los clústeres proporcionan un mecanismo para interconectar gestores de colas de forma que simplifica la configuración inicial y la gestión continua. Puede definir componentes de clúster, y crear y gestionar los clústeres.

“Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms” en la página 95
Puede cambiar el comportamiento de IBM MQ o de un gestor de colas individual para que se ajuste a las necesidades de la instalación editando la información en los archivos de configuración (`.ini`). También puede cambiar las opciones de configuración para IBM MQ MQI clients.

“Configuring queue managers on z/OS” en la página 896
Use these instructions to configure queue managers on IBM MQ for z/OS.

“Setting up communications with other queue managers on z/OS” en la página 975
This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

Técnicas de gestión de colas distribuidas de IBM MQ

Los subtemas de esta sección describen técnicas que se deben utilizar cuando planifique canales. Estos subtemas describen técnicas que le permitirán planificar la forma de conectar los gestores de colas y gestionar el flujo de mensajes entre las aplicaciones.

Para consultar ejemplos de planificación de canal de mensajes, consulte:

- ALW Ejemplo de planificación de canal de mensajes para AIX, Linux, and Windows
- IBM i Ejemplo de planificación de canal de mensajes para IBM i
- z/OS Ejemplo de planificación de canal de mensajes para z/OS
- z/OS Ejemplo de planificación de canal de mensajes para z/OS utilizando grupos de compartición de colas

Conceptos relacionados

Canales
Introducción a la colocación de mensajes en colas
Gestión de colas distribuidas y clústeres

Tareas relacionadas

“Configuración de la gestión de colas distribuidas” en la página 210
En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

Referencia relacionada
Información de configuración de ejemplo

Control de flujo de mensajes

El control de flujo de mensajes es una tarea que implica la configuración y mantenimiento de rutas de mensajes entre gestores de colas. Es importante para rutas que saltan por muchos gestores de colas. En este apartado se describe cómo utilizar colas, definiciones de cola alias y canales de mensajes en el sistema para lograr el control de flujo de mensajes.

Puede controlar el flujo de mensajes utilizando una serie de técnicas que se introdujeron en la "Configuración de la gestión de colas distribuidas" en la página 210. Si el gestor de colas está en un clúster, el flujo de mensajes se controla utilizando técnicas diferentes, tal como se describe en "Control de flujo de mensajes" en la página 212. z/OS Si los gestores de colas están en un grupo de compartición de colas y están habilitadas las colas dentro del grupo (IGQ), entonces el flujo de mensajes se puede controlar mediante los agentes de IGQ. Estos agentes se describen en Transferencia a colas entre grupos.

Puede utilizar los objetos siguientes para lograr el control de flujo de mensajes:

- Colas de transmisión
- Canales de mensajes
- Definición de cola remota
- Definición de alias del gestor de colas
- Definición de alias de cola de respuesta

El gestor de colas y los objetos de cola se describen en Tipos de objeto. Los canales de mensajes se describen en Componentes de gestión de colas distribuidas. Las técnicas siguientes utilizan estos objetos para crear flujos de mensajes en el sistema:

- Transferir mensajes a colas remotas
- Direccionamiento a través de las colas de transmisión específicas
- Recepción de mensajes
- Pasar mensajes a través del sistema
- Separar flujos de mensajes
- Conmutar un flujo de mensajes a otro destino
- Resolver el nombre de cola de respuesta a un nombre de alias

Nota

Todos los conceptos descritos en este apartado son relevantes para todos los nodos en una red, e incluyen el envío y la recepción de extremos de canales de mensajes. Por este motivo, sólo se ilustra un nodo en la mayoría de los ejemplos. La excepción es que el ejemplo requiere la cooperación explícita del administrador en el otro extremo de un canal de mensajes.

Antes de continuar con las técnicas individuales, resulta útil resumir los conceptos de resolución de nombres y las tres formas de utilizar definiciones de colas remotas. Consulte Gestión de colas distribuidas y clústeres.

Conceptos relacionados
"Nombres de colas en cabecera de transmisión" en la página 213
Los nombres de colas de destino viajan con el mensaje en la cabecera de transmisión hasta que se ha alcanzado la cola de destino.

"Cómo crear gestor de colas y alias de respuestas" en la página 213
En este tema se explican las tres maneras de crear una definición de cola remota.

Nombres de colas en cabecera de transmisión

Los nombres de colas de destino viajan con el mensaje en la cabecera de transmisión hasta que se ha alcanzado la cola de destino.

El nombre de cola utilizado por la aplicación, el nombre de cola lógico, lo resuelve el gestor de colas en el nombre de cola de destino. En otras palabras, el nombre de la cola física. Este nombre de cola de destino viaja con el mensaje en un área de datos separada, la cabecera de transmisión, hasta que se ha alcanzado la cola de destino. A continuación, la cabecera de transmisión se elimina.

Cambie la parte del gestor de colas de este nombre de cola al crear clases de servicio paralelas. Recuerde devolver el nombre del gestor de colas al nombre original cuando se haya alcanzado la desviación de la clase de servicio.

Cómo crear gestor de colas y alias de respuestas

En este tema se explican las tres maneras de crear una definición de cola remota.

El objeto de definición de cola remota se utiliza de tres maneras diferentes. Tabla 17 en la página 213 explica cómo definir cada una de las tres maneras:

- Mediante una definición de cola remota para redefinir un nombre de cola local.

  La aplicación sólo proporciona el nombre de cola al abrir una cola y este nombre de cola es el nombre de la definición de cola remota.

  La definición de cola remota contiene los nombres de la cola de destino y del gestor de colas. Opcionalmente, la definición puede contener el nombre de la cola de transmisión que se utilizará. Si no se proporciona el nombre de la cola de transmisión, el gestor de colas utiliza el nombre del gestor de colas, tomado de la definición de cola remota, para el nombre de cola de transmisión. Si no se ha definido una cola de transmisión de este nombre, pero se ha definido una cola de transmisión predeterminada, se utiliza la cola de transmisión predeterminada.
- Mediante una definición de cola remota para redefinir un nombre de gestor de colas.

  La aplicación, o el programa de canal, proporcionan un nombre de cola junto con el nombre del gestor de colas al abrir la cola.

  Si ha proporcionado una definición de cola remota con el mismo nombre que el nombre del gestor de colas y ha dejado el nombre de cola en la definición en blanco, el gestor de colas sustituye el nombre del gestor de colas en la llama abierta con el nombre del gestor de colas en la definición.

  Además, la definición puede contener el nombre de la cola de transmisión que se utilizará. Si no se proporciona el nombre de la cola de transmisión, el gestor de colas adopta el nombre del gestor de colas, tomado de la definición de cola remota, para el nombre de cola de transmisión. Si no se ha definido una cola de transmisión de este nombre, pero se ha definido una cola de transmisión predeterminada, se utiliza la cola de transmisión predeterminada.
- Mediante una definición de cola remota para redefinir un nombre de cola de respuesta.

  Cada vez que una aplicación transfiere un mensaje a una cola, puede proporcionar el nombre de una cola de respuesta para los mensajes de respuesta, pero con el nombre del gestor de colas en blanco.

  Si proporciona una definición de cola remota con el mismo nombre que la cola de respuesta, el gestor de colas local sustituye el nombre de la cola de respuesta por el nombre de la cola de la definición.

  Puede proporcionar un nombre de gestor de colas en la definición, pero no un nombre de cola de transmisión.

Tabla 17. Tres maneras de utilizar el objeto de definición de cola remota

Utilización	**Nombre del gestor de colas**	**Nombre de cola**	**Nombre de cola de transmisión**
1. Definición de cola remota (en la llamada OPEN)			
Se proporciona en la llamada	QM en blanco o local	(*) necesario	no aplicable

Tabla 17. Tres maneras de utilizar el objeto de definición de cola remota (continuación)			
Utilización	**Nombre del gestor de colas**	**Nombre de cola**	**Nombre de cola de transmisión**
Se proporciona en la definición	necesario	necesario	opcional
2. Alias de gestor de colas (en la llamada OPEN)			
Se proporciona en la llamada	(*) necesario y no QM local	necesario	no aplicable
Se proporciona en la definición	necesario	en blanco	opcional
3. Alias de cola de respuesta (en la llamada PUT)			
Se proporciona en la llamada	en blanco	(*) necesario	no aplicable
Se proporciona en la definición	opcional	opcional	en blanco

Nota: (*) significa que este nombre es el nombre del objeto de definición

Para obtener una descripción formal, consulte Resolución de nombres de colas.

Transferir mensajes a colas remotas

Puede utilizar los objetos de definición de cola remota para resolver un nombre de cola en una cola de transmisión para un gestor de colas adyacente.

En un entorno de colas distribuido, una cola y un canal de transmisión son el punto central para todos los mensajes a una ubicación si los mensajes se originan en las aplicaciones en el sistema local, o llegan a través de los canales de un sistema adyacente. La Figura 6 en la página 214 muestra una aplicación que transfiere mensajes a una cola lógica llamada 'QA_norm'. La resolución de nombres utiliza la definición de cola remota 'QA_norm' para seleccionar la cola de transmisión QMB. A continuación, añade una cabecera de transmisión a los mensajes que indican 'QA_norm at QMB'.

Los mensajes que llegan del sistema adyacente en 'Channel_back' tienen una cabecera de transmisión con el nombre de cola física 'QA_norm at QMB', por ejemplo. Estos mensajes se colocan en la cola de transmisión sin QMB.

El canal mueve los mensajes a un gestor de colas adyacente.

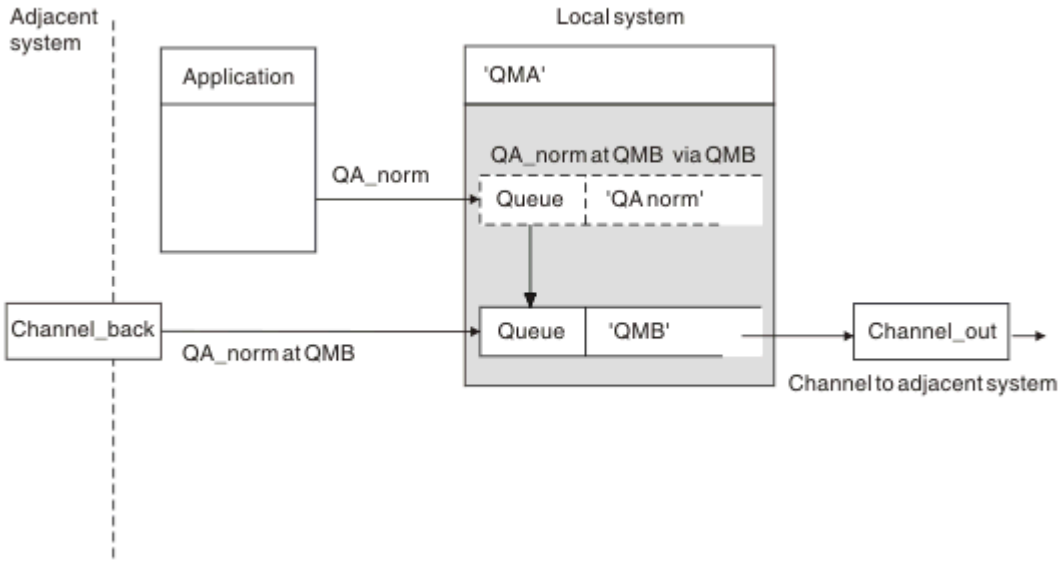


Figura 6. Una definición de cola remota se utiliza para resolver un nombre de cola en una cola de transmisión para un gestor de colas adyacente

Si es el administrador del sistema de IBM MQ, debe:

- Definir el canal de mensajes del sistema adyacente
- Definir el canal de mensajes al sistema adyacente
- Crear la cola de transmisión QMB
- Definir el objeto de cola remota 'QA_norm' para resolver el nombre de cola utilizado por las aplicaciones en el nombre de cola de destino, el nombre de gestor de colas de destino y el nombre de cola de transmisión

En un entorno de clúster, sólo necesita definir un canal de clúster receptor en el gestor de colas local. No es necesario que defina una cola de transmisión o un objeto de cola remoto. Consulte Clústeres.

Más sobre resolución de nombres

El efecto de la definición de cola remota es definir un nombre de cola de destino físico y un nombre de gestor de colas. Estos nombres se colocan en las cabeceras de transmisión de mensajes.

Los mensajes entrantes desde un sistema adyacente ya han tenido este tipo de resolución de nombres realizado por el gestor de colas original. Por lo tanto, tienen la cabecera de transmisión que muestra el nombre de la cola de destino física y el nombre del gestor de colas. Estos mensajes no se ven afectados por las definiciones de colas remotas.

Referencia relacionada
Resolución de nombres de colas

Elección de la cola de transmisión

Puede utilizar una definición de cola remota para permitir que una cola de transmisión diferente envíe mensajes al mismo gestor de colas adyacente.

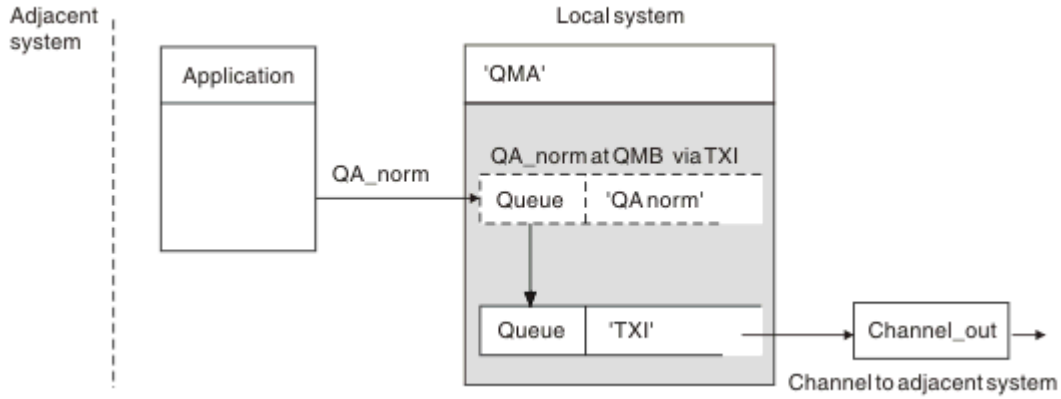


Figura 7. La definición de cola remota permite que se utilice una cola de transmisión diferente

En un entorno de colas distribuido, cuando necesite cambiar un flujo de mensajes de un canal a otro, utilice la misma configuración del sistema que se muestran en la Figura 6 en la página 214 en "Transferir mensajes a colas remotas" en la página 214. La Figura 7 en la página 215 en este tema muestra cómo utilizar la definición de cola remota para enviar mensajes a través de una cola de transmisión distinta, y, por consiguiente, a través de un canal diferente al mismo gestor de colas adyacente.

Para la configuración mostrada en la Figura 7 en la página 215, debe proporcionar el objeto de cola remota 'QA_norm' y la cola de transmisión 'TX1'. Debe proporcionar 'QA_norm' para elegir la cola 'QA_norm' en el gestor de colas remoto, la cola de transmisión 'TX1' y el gestor de colas 'QMB_priority'. Especifique 'TX1' en la definición del canal adyacente al sistema.

Los mensajes se colocan en la cola de transmisión 'TX1' con una cabecera de transmisión que contiene 'QA_norm at QMB_priority' y se envían a través del canal al sistema adyacente.

Channel_back ha quedado fuera de esta ilustración porque necesita un alias de gestor de colas.

En un entorno de clúster, no es necesario que defina una cola de transmisión o una definición de cola remota. Para obtener más información, consulte “Definición de cola de clúster” en la página 310.

Recepción de mensajes

Puede configurar el gestor de colas para recibir mensajes de otros gestores de colas. Debe asegurarse de que no se produzca una resolución de nombres accidental.

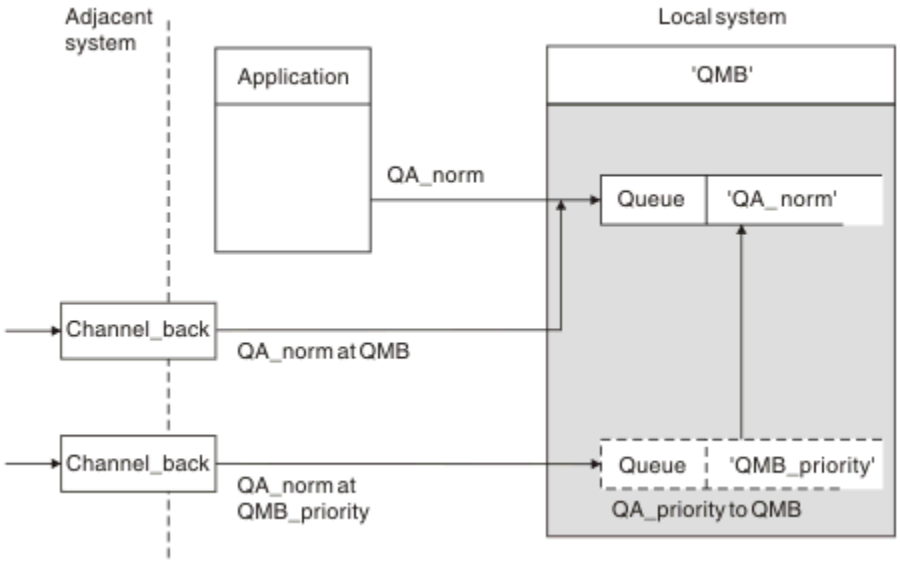


Figura 8. Recepción directa de mensajes y resolución de nombres de alias de gestor de colas

El administrador del sistema debe ocuparse de la organización de los mensajes que se van a enviar así como de los mensajes que se recibirán de los gestores de colas adyacentes. Los mensajes recibidos contienen el nombre físico del gestor de colas de destino y están en cola en la cabecera de transmisión. Se tratan igual como mensajes desde una aplicación local que especifica el nombre del gestor de colas y el nombre de cola. Debido a este tratamiento, debe asegurarse de que los mensajes que entran en el sistema no produzcan una resolución accidental de los nombres. Para este caso, consulte la Figura 8 en la página 216.

Para esta configuración, debe preparar:

- Canales de mensajes que reciban mensajes de los gestores de colas adyacentes
- Una definición del alias de gestor de colas que resuelva un flujo de mensajes de entrada, 'QMB_priority', para el nombre del gestor de colas local, 'QMB'
- La cola local, 'QA_norm', si no existe

Recepción de nombres del gestor de colas alias

El uso de la definición alias de gestor de colas en esta ilustración no ha seleccionado un gestor de colas de destino diferente. Los mensajes que pasan por este gestor de colas local y se dirigen a 'QMB_priority' están pensados para el gestor de colas 'QMB'. El nombre del gestor de colas alias se utiliza para crear el flujo de mensajes por separado.

Pasar mensajes a través del sistema

Puede pasar mensajes por el sistema de tres formas - utilizando el nombre de ubicación, un alias para el gestor de colas o seleccionando una cola de transmisión.

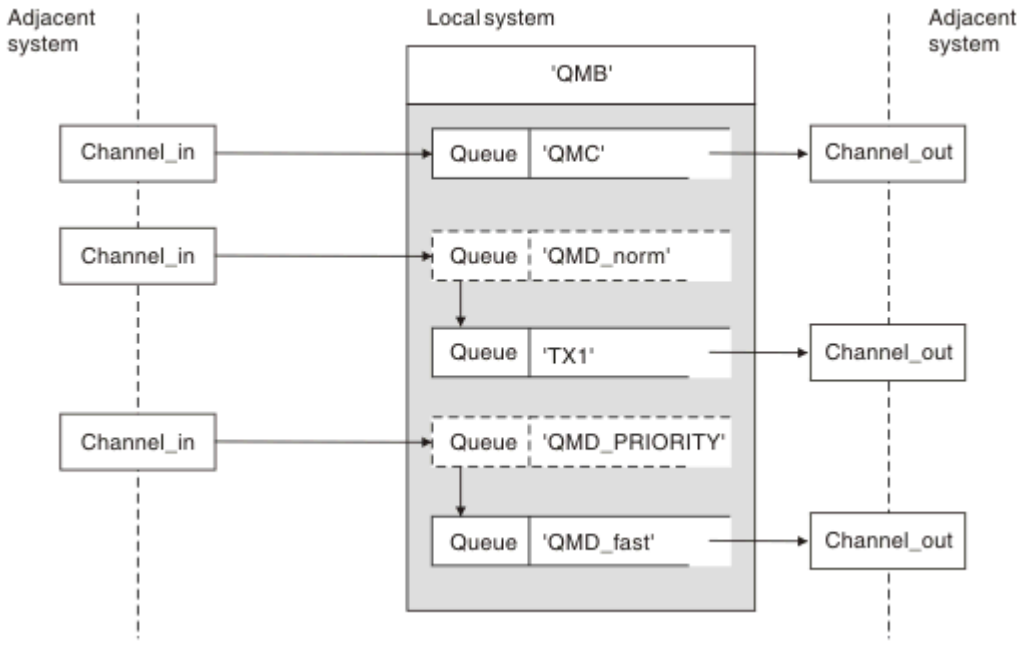


Figura 9. Tres métodos para pasar mensajes por el sistema

La técnica que se muestra en la Figura 8 en la página 216 del apartado "Recepción de mensajes" en la página 216, mostraba cómo se captura un flujo de alias. Figura 9 en la página 217 ilustra los modos en que se forjan las redes aglutinando las técnicas descritas anteriormente.

La configuración muestra un canal que entrega tres mensajes con destinos diferentes:

1. QB en QMC
2. QB en QMD_norm
3. QB en QMD_PRIORITY

Debe pasar el primer flujo de mensajes por el sistema sin modificarlo. Debe pasar el segundo flujo de mensajes a través de un canal y una cola de transmisión diferentes. Para el segundo flujo de mensajes, también debe resolver mensajes para el nombre del gestor de colas de alias QMD_norm en el gestor de colas QMD. El tercer flujo de mensajes opta por una cola de transmisión diferente sin ningún otro cambio.

En un entorno en clúster, los mensajes pasan a través de una cola de transmisión en clúster. Normalmente, una sola cola de transmisión, SYSTEM.CLUSTER.TRANSMIT.QUEUE, transfiere todos los mensajes a todos los gestores de colas de todos los clústeres de los que el gestor de colas es miembro; consulte Un clúster de gestores de colas. Puede definir colas de transmisión separadas para todos los gestores de colas o algunos de ellos en los clústeres de los que es miembro el gestor de colas.

Los métodos siguientes describen técnicas que se pueden aplicar a un entorno de gestión de colas distribuidas.

Utilice estos métodos

Para estas configuraciones, debe preparar:

- Definiciones de canal de entrada
- Definiciones de canal de salida
- Colas de transmisión:

  - QMC
  - TX1
  - QMD_fast
- Definiciones de alias del gestor de colas:
  - QMD_norm con QMD_norm en QMD a través de TX1
  - QMD_PRIORITY con QMD_PRIORITY en QMD_PRIORITY a través de QMD_fast

Nota: Ninguno de los flujos de mensajes que se muestran en el ejemplo cambia la cola de destino. Los alias del gestor de colas proporcionan separación de flujos de mensajes.

Método 1: Utilización del nombre de ubicación entrante

Va a recibir mensajes con una cabecera de transmisión que contiene otro nombre de ubicación como, por ejemplo, QMC. La configuración más sencilla es crear una cola de transmisión con dicho nombre, QMC. El canal que da servicio a la cola de transmisión entrega el mensaje sin modificar al siguiente destino.

Método 2: Utilización de un alias para el gestor de colas

El segundo método es utilizar la definición de objeto de alias del gestor de colas, pero especificar un nuevo nombre de ubicación, QMD y una cola de transmisión específica, TX1. Esta acción:

- Termina el flujo de mensajes alias configurado por el alias de nombre del gestor de colas QMD_norm, es decir, la clase de servicio con el nombre QMD_norm.
- Cambia las cabeceras de transmisión en estos mensajes de QMD_norm a QMD.

Método 3: Selección de una cola de transmisión

El tercer método es tener un objeto alias de gestor de colas definido con el mismo nombre que la ubicación de destino, QMD_PRIORITY. Utilice la definición de alias del gestor de colas para seleccionar una cola de transmisión específica, QMD_fast y, por consiguiente, otro canal. Las cabeceras de transmisión en estos mensajes siguen sin cambios.

Separar flujos de mensajes

Puede utilizar un alias de gestor de colas para crear flujos de mensajes por separado para enviar mensajes al mismo gestor de colas.

Razones para separar mensajes en distintos flujos de mensajes

En un entorno de gestión de colas distribuidas, la necesidad de separar mensajes en el mismo gestor de colas en distintos flujos de mensajes puede deberse a varios motivos. Por ejemplo:

- Es posible que necesite proporcionar un flujo distinto para mensajes grandes, medianos y pequeños. Esta necesidad también se aplica en un entorno en clúster y, en este caso, puede crear clústeres que se solapen. Existe una serie de motivos para ello, por ejemplo:
  - Para permitir que organizaciones diferentes tengan su propia administración.
  - Para permitir que aplicaciones independientes se administren por separado.
  - Para crear una clase de servicio. Por ejemplo, podría tener un clúster llamado PERSONAL que sea un subconjunto del clúster denominado ESTUDIANTES. Al transferir un mensaje a una cola que se anuncia en el clúster PERSONAL, se utiliza un canal restringido. Cuando transfiere un mensaje a una cola anunciada en el clúster ESTUDIANTES, se puede utilizar un canal general o un canal restringido.
  - Para crear entornos de prueba y de producción.
- Es posible que sea necesario direccionar mensajes entrantes por vías de acceso diferentes de la vía de acceso de los mensajes generados localmente.

- Puede que la instalación necesite planificar el movimiento de mensajes en determinados momentos (por ejemplo, durante la noche) y, a continuación, los mensajes deban almacenarse en colas reservadas hasta que se planifiquen.

Flujo de mensajes de ejemplo

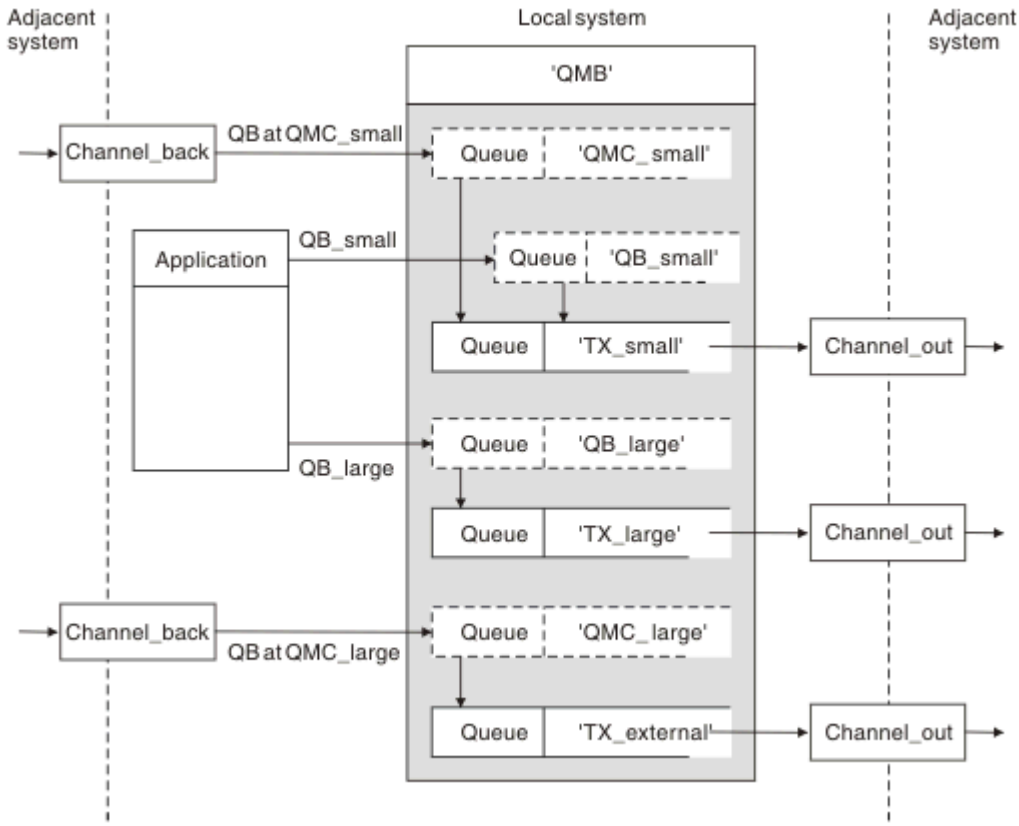


Figura 10. Separación de flujos de mensajes

En el ejemplo que se muestra en la Figura 10 en la página 219, los dos flujos de entrada son los nombres del gestor de colas de alias 'QMC_small' y 'QMC_large'. Estos flujos se proporcionan con una definición de alias de gestor de colas para capturarlos para el gestor de colas local. Dispone de una aplicación que direcciona dos colas remotas y necesita mantener estos flujos de mensaje por separado. Se proporcionan dos definiciones de colas remotas que especifican la misma ubicación, 'QMC', pero especifican diferentes colas de transmisión. Esta definición mantiene los flujos separados, y no se necesita nada adicional en el extremo dado que tienen el mismo nombre de gestor de colas de destino en las cabeceras de transmisión. Se proporcionan:

- Definiciones de canales de entrada
- Las dos definiciones de colas remotas QB_small y QB_large
- Las dos definiciones de alias de gestor de colas remoto QMC_small y QMC_large
- Las tres definiciones de canal emisor
- Tres colas de transmisión: TX_small, TX_large y TX_external

Coordinación con sistemas adyacentes

Cuando se utiliza un alias de gestor de colas para crear un flujo de mensajes separado, necesita coordinar esta actividad con el administrador del sistema en el extremo remoto del canal de mensajes para asegurarse de que el alias de gestor de colas correspondiente esté disponible allí.

Concentración de mensajes en diversas ubicaciones

Puede concentrar mensajes destinados a diversas ubicaciones en un solo canal.

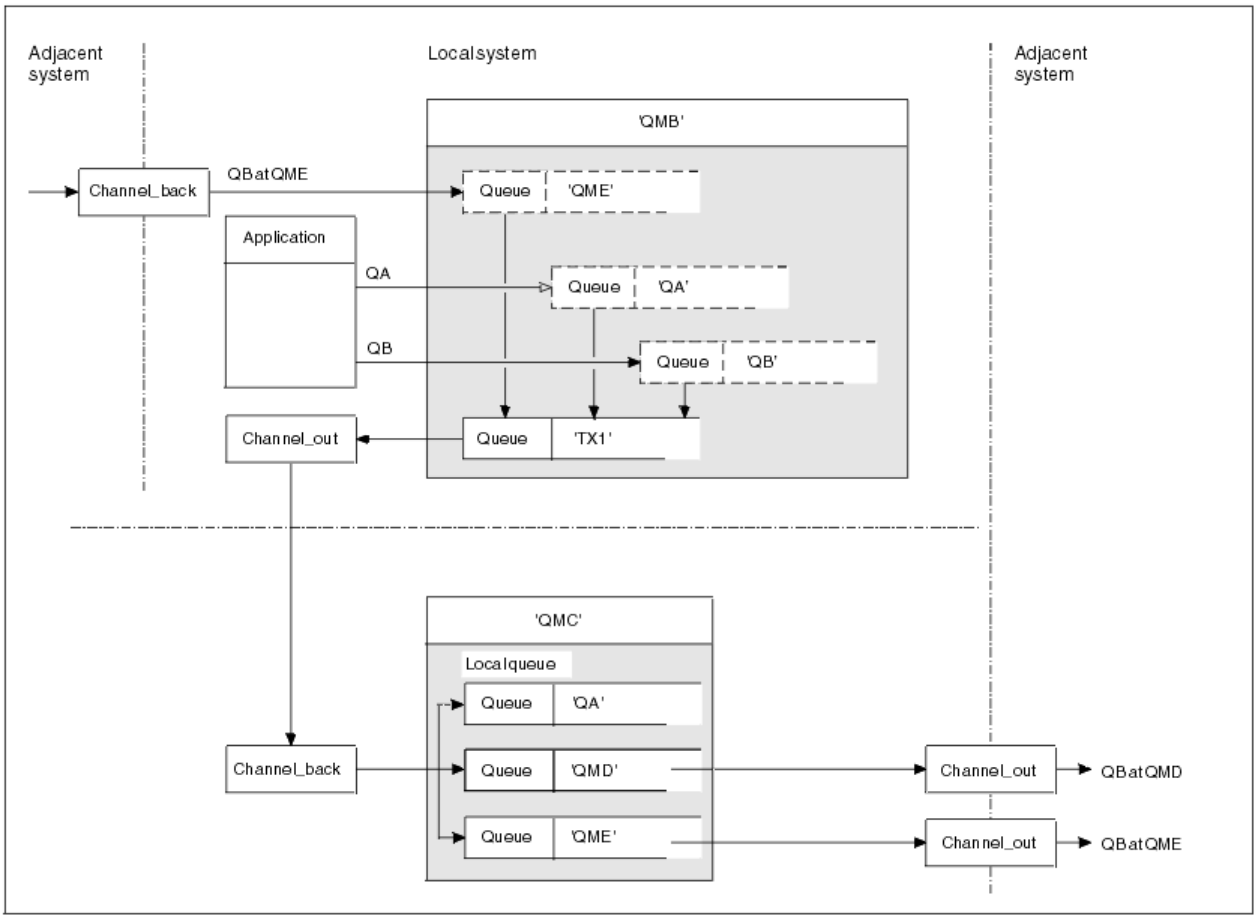


Figura 11. Combinación de flujos de mensajes en un canal

Figura 11 en la página 220 ilustra una técnica de gestión de colas distribuidas para concentrar mensajes que están destinados para diversas ubicaciones en un canal. Dos usos posibles serían:

- Concentración de tráfico de mensajes a través de una pasarela
- Utilización de autopistas de anchos de banda entre nodos

En este ejemplo, los mensajes de diferentes orígenes, locales y adyacentes, y que tienen distintas colas de destino y diferentes gestores de colas, fluyen a través de la cola de transmisión 'TX1' al gestor de colas QMC. El gestor de colas QMC entrega los mensajes de acuerdo con los destinos. Uno definido en una cola de transmisión 'QMD' para la transmisión posterior al gestor de colas QMD. Otro definido en una cola de transmisión 'QME' para la transmisión posterior al gestor de colas QME. Los demás mensajes se colocan en la cola local 'QA'.

Debe proporcionar:

- Definiciones de canal
- Cola de transmisión TX1
- Definiciones de colas remotas:
  - QA con 'QA en QMC a través de TX1'
  - QB con 'QB en QMD a través de TX1'
- Definición de alias de gestor de colas:
  - QME con 'QME a través de TX1'

El administrador complementario que está configurando QMC debe proporcionar:

- Definición de canal receptor con el mismo nombre de canal
- Cola de transmisión QMD con la definición de canal emisor asociada
- Cola de transmisión QME con la definición de canal emisor asociada
- Objeto de cola local QA.

Desvío de flujos de mensajes a otro destino

Puede redefinir el destino de determinados mensajes utilizando los alias de gestor de colas y colas de transmisión.

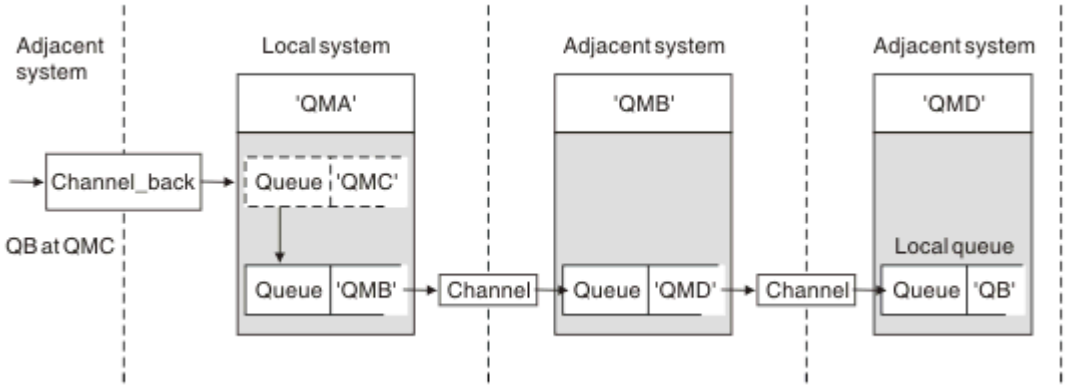


Figura 12. Desvío de corrientes de mensajes a otro destino

Figura 12 en la página 221 ilustra cómo se puede redefinir el destino de determinados mensajes. Los mensajes entrantes a QMA están destinados a 'QB en QMC'. Suelen llegar a QMA y colocarse en una cola de transmisión denominada QMC que ha formato parte de un canal a QMC. QMA debe desviar los mensajes a QMD, pero es capaz de alcanzar QMD sólo a través de QMB. Este método es útil cuando necesita mover un servicio de una ubicación a otra y permite que los suscriptores continúen enviando mensajes en una base temporal hasta que se hayan adaptado a la nueva dirección.

El método de redireccionar mensajes entrantes destinados a un gestor de colas determinado a un gestor de colas diferente utiliza:

- Un alias de gestor de colas para cambiar el gestor de colas de destino a otro gestor de colas y para seleccionar una cola de transmisión al sistema adyacente
- Una cola de transmisión para servir al gestor de colas adyacente
- Una cola de transmisión en el gestor de colas adyacente para direccionamiento posterior al gestor de colas de destino

Debe proporcionar:

- Definición de channel_back
- Alias de gestor de colas QMC con QB en QMD a través de QMB
- Definición de channel_out
- La cola de transmisión asociada QMB

El administrador complementario que está configurando QMB debe proporcionar:

- La definición de channel_back correspondiente
- La cola de transmisión, QMD
- La definición de canal asociado a QMD

Puede utilizar alias en un entorno en clúster. Para obtener información, consulte “Alias de gestor de colas y clústeres” en la página 406.

Envío de mensajes a una lista de distribución

En Multiplatforms, puede utilizar una sola llamada MQPUT para que una aplicación envíe un mensaje a varios destinos.

En IBM MQ en Multiplatforms, una aplicación puede enviar un mensaje a varios destinos con una sola llamada MQPUT. Puede hacerlo en un entorno de gestión de colas distribuidas y un entorno en clúster. Deberá definir los destinos en una lista de distribución, tal como se describe en Listas de distribución.

No todos los gestores de colas dan soporte a listas de distribución. Cuando un MCA establece una conexión con un socio, determina si el socio da soporte a listas de distribución y establece un distintivo en la cola de transmisión en consecuencia. Si una aplicación intenta enviar un mensaje que está destinado a una lista de distribución pero el socio no da soporte a listas de distribución, el MCA emisor intercepta el mensaje y lo transfiere a la cola de transmisión una vez para cada destino previsto.

Un MCA receptor asegura que los mensajes enviados a una lista de distribución son recibidos de forma segura en todos los destinos previstos. Si los destinos fallan, el MCA establece los que han fallado. A continuación, puede generar informes de excepción para ellos y puede intentar enviarles los mensajes de nuevo.

Cola de respuestas

Puede crear un bucle de proceso de cola remota completo utilizando una cola de respuesta.

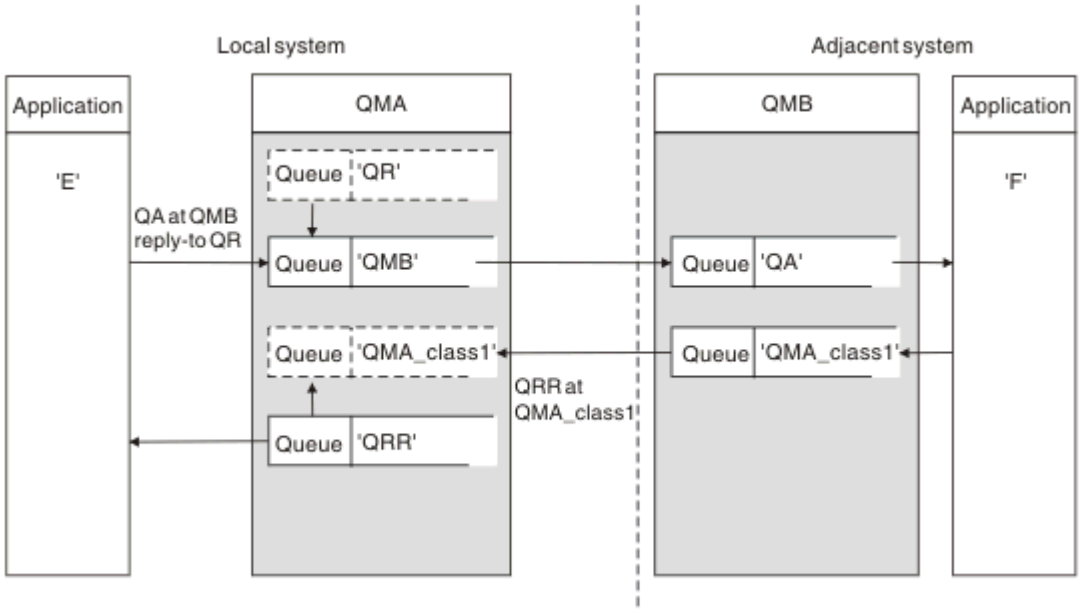


Figura 13. Sustitución de nombre de cola de respuesta durante una llamada PUT

En la Figura 13 en la página 222 se muestra un bucle de proceso de cola remota completo utilizando una cola de respuesta. Este bucle se aplica a un entorno de gestión de colas distribuidas y un entorno en clúster. Los detalles aparecen en la Tabla 21 en la página 230.

La aplicación abre QA en QMB y transfiere mensajes en dicha cola. Se da a los mensajes un nombre de cola de respuesta de QR, sin que se especifique el nombre del gestor de colas. El QMA del gestor de colas busca el objeto de cola de respuesta QR y le extrae el nombre de alias de QRR y el nombre del gestor de colas QMA_class1. Estos nombres se transfieren a los campos de respuesta de los mensajes.

Los mensajes de respuesta de las aplicaciones en QMB están dirigidos a QRR en QMA_class1. La definición del nombre de alias del gestor de clase QMA_class1 lo utiliza el gestor de colas para que los mensajes fluyan a si mismo y a la cola QRR.

Este escenario ilustra el modo en que da a las aplicaciones la posibilidad de elegir una clase de servicio para mensajes de respuesta. La clase está implementada por la cola de transmisión QMA_class1 en QMB, junto con la definición de alias del gestor de colas, QMA_class1 en QMA. De este modo, puede cambiar la

cola de respuesta de una aplicación para que los flujos se separen sin que esto implique a la aplicación. La aplicación siempre elige QR para esta clase de servicio específica. Tiene la oportunidad de cambiar la clase de servicio con la definición de cola de respuesta QR.

Debe crear:

- Definición de cola de respuesta QR
- Objeto de cola de transmisión QMB
- Definición de channel_out
- Definición de channel_back
- Definición de alias del gestor de colas QMA_class1
- Objeto de cola local QRR, si no existe

El administrador complementario en el sistema adyacente debe crear:

- Definición de canal receptor
- Objeto de cola de transmisión QMA_class1
- Canal emisor asociado
- Objeto de cola local QA.

Los programas de la aplicación utilizan:

- Nombre de cola de respuesta QR en llamadas put
- Nombre de cola QRR en llamadas get

De este modo, puede cambiar la clase de servicio según sea necesario, sin que implique la aplicación. Cambia el alias de respuesta 'QR', junto con la cola de transmisión 'QMA_class1' y el alias del gestor de colas 'QMA_class1'.

Si no se encuentra ningún objeto alias de respuesta cuando se transfiere el mensaje a la cola, el nombre del gestor de colas local se inserta en el campo del nombre del gestor de colas de respuesta en blanco. El nombre de la cola de respuesta permanece sin cambios.

Restricción de resolución de nombres

Debido a que la resolución de nombres se ha llevado a cabo para la cola de respuesta en 'QMA' cuando se transfirió el mensaje original, no se permite ninguna resolución de nombres más en 'QMB'. El mensaje lo transfiere con el nombre físico de la cola de respuesta la aplicación que responde.

Las aplicaciones deben saber que el nombre que utilizan para la cola de respuesta es diferente del nombre de la cola real donde se encuentran los mensajes de retorno.

Por ejemplo, cuando se proporcionan dos clases de servicio para el uso de aplicaciones con nombres de alias de colas de respuesta de 'C1_alias' y 'C2_alias', las aplicaciones utilizan estos nombres como nombres de colas de respuesta en las llamadas put del mensaje. No obstante, las aplicaciones esperan en realidad que aparezcan mensajes en las colas 'C1' para 'C1_alias' y 'C2' para 'C2_alias'.

Sin embargo, una aplicación es capaz de realizar una llamada de consulta en la cola alias de respuesta para comprobar por sí misma el nombre de la cola real que debe utilizar para obtener los mensajes de respuesta.

Conceptos relacionados

“Cómo crear gestor de colas y alias de respuestas” en la página 213
En este tema se explican las tres maneras de crear una definición de cola remota.

“Ejemplo de alias de cola de respuesta” en la página 224
En este ejemplo se muestra el uso de un alias de respuesta para seleccionar una ruta diferente (cola de transmisión) para los mensajes devueltos. El uso de este recurso requiere que el nombre de la cola de respuesta cambie en cooperación con las aplicaciones.

“Cómo funciona el ejemplo” en la página 225
Una explicación del ejemplo y de cómo el gestor de colas utiliza el alias de la cola de respuesta.

“Recorrido del alias de colas de respuesta” en la página 226
Un recorrido por el proceso desde el momento en que la aplicación transfiere un mensaje a una cola remota hasta que la misma aplicación elimina el mensaje de respuesta de la cola alias de respuesta.

Ejemplo de alias de cola de respuesta

En este ejemplo se muestra el uso de un alias de respuesta para seleccionar una ruta diferente (cola de transmisión) para los mensajes devueltos. El uso de este recurso requiere que el nombre de la cola de respuesta cambie en cooperación con las aplicaciones.

Tal como se muestra en la Figura 14 en la página 224, la ruta de retorno debe estar disponible para los mensajes de respuesta, incluida la cola de transmisión, el canal y el alias de gestor de colas.

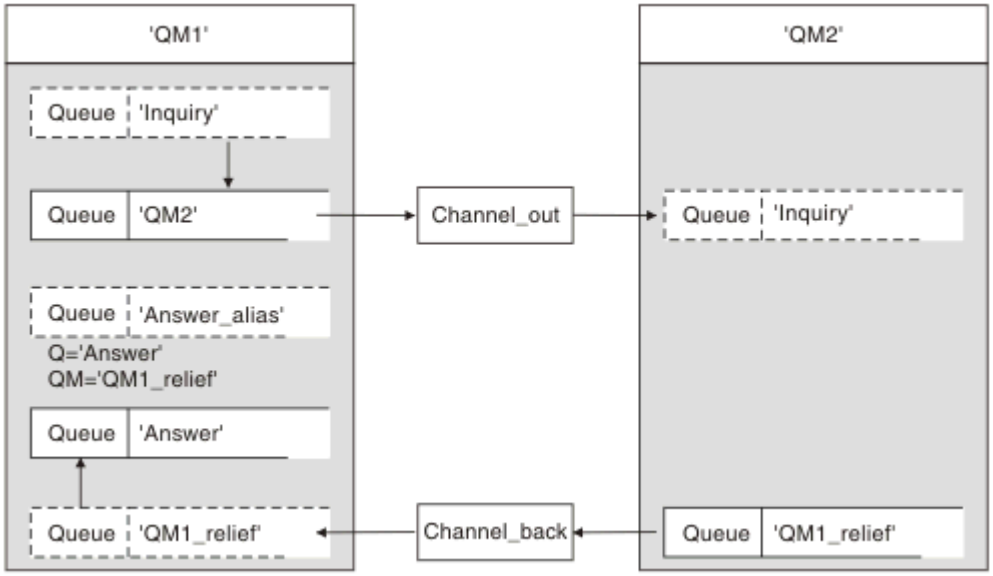


Figura 14. Ejemplo de alias de cola de respuesta

Este ejemplo es para las aplicaciones del peticionario en 'QM1' que envían mensajes a aplicaciones del servidor en 'QM2'. Los mensajes en el servidor se devolverán a través de un canal alternativo utilizando la cola de transmisión 'QM1_relief' (el canal de retorno predeterminado se servirá con una cola de transmisión 'QM1').

El alias de cola de respuesta es un uso específico de la definición de cola remota denominada 'Answer_alias'. Las aplicaciones en QM1 incluyen este nombre, 'Answer_alias', en el campo de respuesta de todos los mensajes que transfieren a la cola 'Inquiry'.

La definición de colas de respuesta 'Answer_alias' está definida como 'Answer en QM1_relief'. Las aplicaciones en QM1 esperan que las respuestas aparezcan en la cola local denominada 'Answer'.

Las aplicaciones de servidor en QM2 utilizan el campo de respuesta de los mensajes recibidos para obtener los nombres del gestor de colas y de la cola para los mensajes de respuesta al solicitante en QM1.

Definiciones utilizadas en este ejemplo en QM1

El administrador del sistema de IBM MQ en QM1 debe asegurarse de que la cola de respuesta 'Answer' se crea junto con los demás objetos. El nombre del alias del gestor de colas, marcado con un '*', debe coincidir con el nombre del gestor de colas de la definición de alias de cola de respuesta, también marcado con un '*'.

Objeto	**Definición**	
Cola de transmisión local	QM2	
Definición de cola remota	Nombre de objeto	Consulta

Objeto	Definición	
	Nombre del gestor de colas remoto	QM2
	Nombre de cola remota	Consulta
	Nombre de cola de transmisión	QM2 (DEFAULT)
Alias del gestor de colas	Nombre de objeto	QM1_relief *
	Nombre del gestor de colas	QM1
	Nombre de cola	(en blanco)
Alias de cola de respuesta	Nombre de objeto	Answer_alias
	Nombre del gestor de colas remoto	QM1_relief *
	Nombre de cola remota	Respuesta

Definición put en QM1

Las aplicaciones llenan los campos de respuesta con el nombre de alias de cola de respuesta y dejan el campo de nombre del gestor de colas en blanco.

Campo	Contenido
Nombre de cola	Consulta
Nombre del gestor de colas	(en blanco)
Nombre de cola de respuestas	Answer_alias
Gestor de colas de respuestas	(en blanco)

Definiciones utilizadas en este ejemplo en QM2

El administrador del sistema de IBM MQ en QM2 debe asegurarse de que la cola local existe para los mensajes entrantes y de que la cola de transmisión correctamente especificada está disponible para los mensajes de respuestas.

Objeto	Definición
Cola local	Consulta
Cola de transmisión	QM1_relief

Definición put en QM2

Las aplicaciones en QM2 recuperan el nombre de la cola de respuesta y el nombre del gestor de colas del mensaje original y las utilizan al transferir el mensaje de respuesta en la cola de respuesta.

Campo	Contenido
Nombre de cola	Respuesta
Nombre del gestor de colas	QM1_relief

Cómo funciona el ejemplo

Una explicación del ejemplo y de cómo el gestor de colas utiliza el alias de la cola de respuesta.

En este ejemplo, las aplicaciones del peticionario en QM1 siempre utilizan 'Answer_alias' como cola de respuesta en el campo relevante de la llamada put. Siempre recuperan sus mensajes de la cola llamada 'Answer'.

Las definiciones del alias de la cola de respuesta siempre están disponibles para ser utilizadas por el administrador del sistema QM1 para cambiar el nombre de la cola de respuesta 'Answer', y de la ruta de retorno 'QM1_relief'.

Modificar el nombre de cola 'Answer' normalmente no suele ser útil porque las aplicaciones QM1 están esperando sus respuestas en esta cola. No obstante, el administrador del sistema QM1 puede cambiar la ruta de retorno (clase de servicio), según sea necesario.

Cómo el gestor de colas utiliza el alias de cola de respuesta

El gestor de colas QM1 recupera las definiciones del alias de cola de respuesta cuando el nombre de cola de respuesta, incluido en la llamada put realizada por la aplicación, es el mismo que el alias de cola de respuesta y la parte del gestor de colas está en blanco.

El gestor de colas sustituye el nombre de la cola de respuesta en la llamada put por el nombre de cola de la definición. Sustituye el nombre del gestor de colas en blanco en la llamada put por el nombre de gestor de colas de la definición.

Estos nombres se incluyen con el mensaje en el descriptor de mensajes.

Tabla 18. Alias de cola de respuesta

Nombre de campo	**Llamada put**	**Cabecera de transmisión**
Nombre de cola de respuestas	Answer_alias	Respuesta
Nombre del gestor de colas de respuesta	(en blanco)	QM1_relief

Recorrido del alias de colas de respuesta

Un recorrido por el proceso desde el momento en que la aplicación transfiere un mensaje a una cola remota hasta que la misma aplicación elimina el mensaje de respuesta de la cola alias de respuesta.

Para completar este ejemplo, vamos a examinar el proceso.

1. La aplicación abre una cola llamada 'Inquiry', y le transfiere mensajes. La aplicación establece los campos de respuesta del descriptor de mensajes en:

Nombre de cola de respuestas	**Answer_alias**
Nombre del gestor de colas de respuesta	(en blanco)

2. El gestor de colas 'QM1' responde al nombre del gestor de colas en blanco comprobando una definición de cola remota con el nombre 'Answer_alias'. Si no se encuentra ninguno, el gestor de colas coloca su propio nombre, 'QM1', en el campo del gestor de colas de respuesta del descriptor de mensaje.
3. Si el gestor de colas encuentra una definición de cola remota con el nombre 'Answer_alias', extrae el nombre de la cola y los nombres de gestor de colas de la definición (nombre de cola='Answer' y nombre de gestor de colas= 'QM1_relief'). A continuación, los transfiere a los campos de respuesta del descriptor de mensaje.
4. El gestor de colas 'QM1' utiliza la definición de cola remota 'Inquiry' para determinar que la cola de destino prevista está en el gestor de colas 'QM2' y el mensaje se coloca en la cola de transmisión 'QM2'. 'QM2' es el nombre de cola de transmisión predeterminado para los mensajes destinados a colas en el gestor de colas 'QM2'.
5. Cuando el gestor de colas 'QM1' transfiere el mensaje en la cola de transmisión, añade una cabecera de transmisión al mensaje. Esta cabecera contiene el nombre de la cola de destino, 'Inquiry', y el gestor de colas de destino, 'QM2'.
6. El mensaje llega al gestor de colas 'QM2', y se transfiere en la cola local 'Inquiry'.
7. Una aplicación obtiene el mensaje de esta cola y lo procesa. La aplicación prepara un mensaje de respuesta y lo transfiere al nombre de cola de respuesta del descriptor de mensaje del mensaje original:

Nombre de cola de respuestas	Respuesta
Nombre del gestor de colas de respuesta	QM1_relief

8. El gestor de colas 'QM2' realiza el mandato put. La búsqueda del nombre del gestor de colas, 'QM1_relief', es un gestor de colas remoto, coloca el mensaje en la cola de transmisión con el mismo nombre, 'QM1_relief'. Al mensaje se le proporciona una cabecera de transmisión que contiene el nombre de la cola de destino, 'Answer', y el gestor de colas de destino, 'QM1_relief'.
9. El mensaje se transfiere al gestor de colas 'QM1'. El gestor de colas, que reconoce que el nombre del gestor de colas 'QM1_relief' es un alias, extrae de la definición del alias 'QM1_relief' el nombre del gestor de colas físico 'QM1'.
10. A continuación, el gestor de colas 'QM1' coloca el mensaje en el nombre de la cola incluida en la cabecera de transmisión, 'Answer'.
11. La aplicación extrae su mensaje de respuesta de la cola 'Answer'.

Consideraciones de red

En un entorno de colas distribuido, dado que se los destinos de los mensajes se resuelven con únicamente un nombre de cola y un nombre de gestor de colas, se aplican determinadas reglas.

1. Donde se proporciona el nombre del gestor de colas y el nombre es diferente del nombre del gestor de colas local:
   - Una cola de transmisión debe estar disponible con el mismo nombre. Esta cola de transmisión debe ser parte de un canal de mensajes que transfiere mensajes a otro gestor de colas, o bien
   - Una definición de alias de gestor de colas debe existir para resolver el nombre del gestor de colas en el mismo nombre o en otro nombre de gestor de colas y la cola de transmisión opcional, o bien
   - Si el nombre de la cola de transmisión no se puede resolver y se ha definido una cola de transmisión predeterminada, se utiliza la cola de transmisión predeterminada.
2. En el caso de que sólo se suministre el nombre de cola, debe estar disponible una cola de cualquier tipo pero con el mismo nombre en el gestor de colas local. Esta cola puede ser una definición de cola remota que se resuelve en: una cola de transmisión para un gestor de colas adyacente, un nombre de gestor de colas y una cola de transmisión opcional.

Para ver cómo funciona en un entorno de clúster, consulte Clústeres.

z/OS Si los gestores de colas se ejecutan en un grupo de compartición de colas (QSG) y está habilitada una transferencia a colas entre grupos (IGQ), debe utilizar SYSTEM.QSG.TRANSMIT.QUEUE. Para obtener más información, consulte Transferencia a colas entre grupos.

Suponga el caso de un canal de mensajes que transfiere mensajes de un gestor de colas a otro en un entorno de colas distribuido.

Los mensajes que se transfieren provienen de cualquier otro gestor de colas en la red y puede suceder que algunos mensajes tengan un nombre de gestor de colas desconocido como destino. Este problema se puede producir cuando un nombre de gestor de colas ha cambiado o se ha eliminado del sistema, por ejemplo.

El programa de canal reconoce esta situación cuando no puede encontrar una cola de transmisión para estos mensajes y los coloca en la cola de mensajes no entregados. Corresponde al usuario buscar estos mensajes y ocuparse de que se reenvíen al destino correcto. O bien, devolverlos al originador, donde se puede determinar el originador.

Los informes de excepciones se generan en estas circunstancias, si se solicitaron mensajes de informe en el mensaje original.

Convenio de resolución de nombres

La resolución del nombre que cambia la identidad de la cola de destino (es decir, cambio de nombre lógico a físico), sólo se produce una vez y sólo en el gestor de colas de origen.

La utilización posterior de varias posibilidades de alias sólo se debe utilizar al separar y combinar flujos de mensajes.

Direccionamiento de retorno

Los mensajes pueden contener una dirección de retorno en forma de nombre de una cola y un gestor de colas. Este formato de dirección de retorno se puede utilizar en un entorno de colas distribuido y en un entorno en clúster.

Esta dirección la suele especificar la aplicación que crea el mensaje. La puede modificar cualquier aplicación que luego maneja el mensaje, incluidas aplicaciones de salida de usuario.

Independientemente del origen de esta dirección, cualquier aplicación que maneje el mensaje puede elegir utilizar esta dirección para devolver mensajes de respuesta, de estado o de informe a la aplicación de origen.

El modo en que se direccionan estos mensajes no difiere del modo en que se direcciona el mensaje original. Tenga en cuenta que los flujos de mensajes que cree a otros gestores de colas necesitan flujos de retorno correspondiente.

Conflictos de nombres físicos

El nombre de la cola de respuesta se ha resuelto en un nombre de la cola física en el gestor de colas original. No se debe volver a resolver en el gestor de colas de respuesta.

Es una posible fuente de problemas de conflicto de nombres que sólo se puede evitar mediante un acuerdo que abarque toda la red sobre nombres de colas físicas y lógicas.

Gestión de traducciones de nombres de colas

Cuando crea una definición de alias de gestor de colas o una definición de cola remota, la resolución de nombres se realiza para cada mensaje que lleva dicho nombre. Esta situación se debe gestionar.

Esta descripción se proporciona para los diseñadores de aplicaciones y planificadores de canal que traten un sistema individual que tenga canales de mensajes a sistemas adyacentes. Requiere un punto de vista local del control y la planificación del canal.

Cuando crea una definición de alias de gestor de colas o una definición de cola remota, la resolución del nombre se realiza para cada mensaje que lleva dicho nombre, independientemente del origen del mensaje. Para supervisar esta situación, que podría implicar un gran número de colas en una red de gestores de colas, debe mantener tablas de:

- Los nombres de colas de origen y de los gestores de colas de origen con respecto a los nombres de colas resueltos, los nombres de gestores de colas resueltos y los nombres de colas de transmisión resueltos, con método de resolución
- Los nombres de colas de origen con respecto a:
  - Nombres de colas de destino resueltos
  - Nombres de gestores de colas de destino resueltos
  - Colas de transmisión
  - Nombres del canal de mensajes
  - Nombres del sistema adyacente
  - Nombres de colas de respuesta

Nota: El uso del término *origen* en este contexto se refiere al nombre de cola o al nombre del gestor de colas que proporciona la aplicación, o bien a un programa de canal cuando abre una cola para transferir mensajes.

Un ejemplo de cada una de estas tablas se muestra en la Tabla 19 en la página 229, la Tabla 20 en la página 229 y la Tabla 21 en la página 230.

Los nombres de estas tablas se derivan de los ejemplos de esta sección, y esta tabla no está pensada como un ejemplo práctico de la resolución de nombres de colas en un nodo.

Tabla 19. Resolución del nombre de la cola en el gestor de colas QMA

Cola de origen especificada cuando se abre la cola	Gestor de colas de origen especificado cuando se abre la cola	Nombre de cola resuelto	Nombre del gestor de colas resuelto	Nombre de cola de transmisión resuelto	Tipo de resolución
QA_norm	-	QA_norm	QMB	QMB	Cola remota
(cualquiera)	QMB	-	-	QMB	(ninguno)
QA_norm	-	QA_norm	QMB	TX1	Cola remota
QB	QMC	QB	QMD	QMB	Alias del gestor de colas

Tabla 20. Resolución de nombre de cola en el gestor de colas QMB

Cola de origen especificada cuando se abre la cola	Gestor de colas de origen especificado cuando se abre la cola	Nombre de cola resuelto	Nombre del gestor de colas resuelto	Nombre de cola de transmisión resuelto	Tipo de resolución
QA_norm	-	QA_norm	QMB	-	(ninguno)
QA_norm	QMB	QA_norm	QMB	-	(ninguno)
QA_norm	QMB_PRIORITY	QA_norm	QMB	-	Alias del gestor de colas
(cualquiera)	QMC	(cualquiera)	QMC	QMC	(ninguno)
(cualquiera)	QMD_norm	(cualquiera)	QMD_norm	TX1	Alias del gestor de colas
(cualquiera)	QMD_PRIORITY	(cualquiera)	QMD_PRIORITY	QMD_fast	Alias del gestor de colas
(cualquiera)	QMC_small	(cualquiera)	QMC_small	TX_small	Alias del gestor de colas
(cualquiera)	QMC_large	(cualquiera)	QMC_large	TX_external	Alias del gestor de colas
QB_small	QMC	QB_small	QMC	TX_small	Cola remota
QB_large	QMC	QB_large	QMC	TX_large	Cola remota
(cualquiera)	QME	(cualquiera)	QME	TX1	Alias del gestor de colas
QA	QMC	QA	QMC	TX1	Cola remota
QB	QMD	QB	QMD	TX1	Cola remota

Tabla 21. Conversión de nombre de cola de respuesta en el gestor de colas QMA			
Diseño de aplicaciones		**Definición de alias de respuestas**	
QMGR local	**Nombre de cola para mensajes**	**Nombre de alias de cola de respuesta**	**Redefinido en**
QMA	QRR	QR	QRR en QMA_class1

Numeración de secuencia de mensajes de canal

El canal utiliza números de secuencia para comprobar que los mensajes se entregan en el mismo orden en que se toman de la cola de transmisión.

Los números de secuencia de canal se comprueban cuando se inicia un canal y si se produce una discrepancia, implica que los datos de sincronización persistentes se han perdido en cualquiera de los lados del canal; por ejemplo, una configuración de recuperación tras desastre (DR), o que el final del proceso por lotes se ha interrumpido cuando el canal estaba pendiente.

Al restablecer o ignorar las discrepancias de número de secuencia, consulte **IgnoreSeqNumberMismatch** en la sección *Canales del archivo qm.ini*, no corre el riesgo de perder o duplicar un lote de mensajes y no restablece el estado pendiente de un canal.

Esta información se puede visualizar utilizando DISPLAY CHSTATUS. El número de secuencia y un identificador denominado el LUWID se almacenan en el almacenamiento persistente para el último mensaje transferido en un lote. Estos valores se utilizan durante el inicio del canal para asegurarse de que los dos extremos del enlace estén de acuerdo en qué mensajes se han transferido correctamente.

Recuperación secuencial de mensajes

Si una aplicación transfiere una secuencia de mensajes a la misma cola de destino, esos mensajes se pueden recuperar en secuencia mediante una ***única*** aplicación con una secuencia de operaciones MQGET, si se cumplen las siguientes condiciones:

- Todas las solicitudes de transferencia se han realizado a partir de la misma aplicación.
- Todas las solicitudes de transferencia eran de la misma unidad de trabajo o todas las solicitudes de transferencia se han realizado fuera de una unidad de trabajo.
- Los mensajes tienen todos la misma prioridad.
- Los mensajes tienen todos la misma persistencia.
- Para la gestión de colas remotas, la configuración es tal que sólo puede haber una vía de acceso desde la aplicación que realiza la solicitud de transferencia, a través de su gestor de colas, a través de la intercomunicación, hasta el gestor de colas de destino y la cola de destino.
- Los mensajes no se transfieren a la cola de mensajes no entregados (por ejemplo, si una cola está llena temporalmente).
- La aplicación que obtiene el mensaje no cambia deliberadamente el orden de recuperación, por ejemplo, especificando un determinado *MsgId* o *CorrelId* o utilizando prioridades de mensajes.
- Sólo una aplicación está realizando operaciones get para recuperar los mensajes de la cola de destino. Si existe más de una aplicación, estas aplicaciones deben estar diseñadas para obtener todos los mensajes en cada transferencia de secuencia mediante una aplicación de envío.

Nota: Se pueden intercalar mensajes de otras tareas y unidades de trabajo con la secuencia, incluso cuando la secuencia no se transfirió desde dentro de una sola unidad de trabajo.

Si estas condiciones no se pueden cumplir y el orden de los mensajes en la cola de destino es importante, la aplicación se puede codificar para utilizar su propio número de secuencia de mensaje como parte del mensaje para garantizar el orden de los mensajes.

Secuencia de recuperación de mensajes rápidos y no persistentes

Los mensajes no persistentes de un canal rápido pueden superar a los mensajes persistentes en el mismo canal y así llegar fuera de secuencia. El MCA receptor transfiere los mensajes no persistentes a la cola de destino inmediatamente y los hace visible. Los mensajes persistentes no son visibles hasta el siguiente punto de sincronización.

Prueba de bucle de retorno

La *prueba de bucle de retorno* es una técnica en Multiplatforms que le permite probar un enlace de comunicaciones sin enlazar realmente con otra máquina.

Configure una conexión entre dos gestores de colas como si estuvieran en dos máquinas separadas, pero pruebe la conexión mediante un bucle a otro proceso en la misma máquina. Esta técnica significa que puede probar el código de comunicaciones sin necesidad de una red activa.

El modo de hacerlo depende de los productos y protocolos que esté utilizando.

En sistemas Windows, puede utilizar el adaptador de "bucle invertido".

Consulte la documentación de los productos que está utilizando para obtener más información.

Rastreo de la ruta y registro de la actividad

Puede confirmar la ruta que realiza un mensaje a través de una serie de gestores de colas de dos maneras.

Puede utilizar la aplicación de visualización de ruta de IBM MQ, disponible a través del mandato de control **dspmqrte**, o bien puede utilizar el registro de la actividad. Estos dos temas se describen en Referencia de supervisión.

Introducción a la gestión de colas distribuidas

La gestión de colas distribuidas (DQM) se utiliza para definir y controlar la comunicación entre los gestores de colas.

La gestión de colas distribuidas:

- Le permite definir y controlar los canales de comunicación entre los gestores de colas
- Le proporciona un servicio de canal de mensajes para mover mensajes de un tipo de *cola local*, conocido como cola de transmisión, a enlaces de comunicación de un sistema local, y de enlaces de comunicación a colas locales de un gestor de colas de destino
- Le proporciona recursos de supervisión del funcionamiento de los canales y diagnosis de problemas, mediante paneles, mandatos y programas

Las definiciones de canal asocian nombres de canal con colas de transmisión, identificadores de enlaces de comunicación y atributos de canal. Las definiciones de canal se implementan de formas diferentes en cada plataforma. El envío y recepción de mensajes está controlado por programas conocidos como *agentes de canal de mensajes* (MCA), que utilizan las definiciones de canal para iniciar y controlar la comunicación.

Los MCA, a su vez, están controlados por el propio DQM. La estructura depende de la plataforma, pero normalmente incluye escuchas y supervisores desencadenantes, además de mandatos de operador y paneles.

Un *canal de mensajes* es una conexión unidireccional para transferir mensajes de un gestor de colas a otro. De este modo, un canal de mensajes tiene dos puntos finales, representados por un par de MCA. Cada punto final tiene una definición de su extremo del canal de mensajes. Por ejemplo, un extremo podría definir un emisor y el otro extremo un receptor.

Para obtener más detalles sobre cómo definir canales, consulte:

- ALW “Supervisión y control de canales en AIX, Linux, and Windows” en la página 263

- z/OS “Monitoring and controlling channels on z/OS” en la página 979
- IBM i “Supervisión y control de canales en IBM i” en la página 287

Para consultar ejemplos de planificación de canal de mensajes, consulte:

- ALW Ejemplo de planificación de canal de mensajes para AIX, Linux, and Windows
- IBM i Ejemplo de planificación de canal de mensajes para IBM i
- z/OS Ejemplo de planificación de canal de mensajes para z/OS
- z/OS Ejemplo de planificación de canal de mensajes para z/OS utilizando grupos de compartición de colas

Para obtener más información sobre las salidas de canal, consulte Programas de salida de canal para canales de mensajería.

Conceptos relacionados

“Envío y recepción de mensajes” en la página 232
La figura siguiente muestra el modelo de gestión de colas distribuidas, que detalla las relaciones entre entidades cuando se transmiten mensajes. También muestra el flujo de control.

“Función de control de canales” en la página 240
La función de control de canales proporciona recursos para definir, supervisar y controlar canales.

“¿Qué sucede cuando no puede entregarse un mensaje?” en la página 254
Cuando un mensaje no puede entregarse, el MCA puede procesarlo de varias formas. Puede intentarlo de nuevo, puede devolvérselo al emisor o puede ponerlo en la cola de mensajes no entregados.

“Archivos de inicialización y configuración” en la página 259
El manejo de los datos de inicialización del canal depende de la plataforma de IBM MQ.

“Conversión de datos para mensajes” en la página 260
Los mensajes de IBM MQ podrían necesitar la conversión de datos cuando se envían entre colas en distintos gestores de colas.

“Escribir sus propios agentes de canales de mensajes” en la página 260
IBM MQ le permite escribir sus propios programas de agente de canal de mensajes (MCA) o instalar el de un proveedor de software independiente.

“Otras cosas que hay que tener en cuenta para gestionar colas distribuidas” en la página 261
Otros temas que hay que tener en cuenta cuando se prepara IBM MQ para la gestión de colas distribuidas. Este tema cubre las colas de mensajes no entregados, las colas en uso, las extensiones del sistema y los programas de salida de usuario, y la ejecución de canales y escuchas como aplicaciones de confianza.

Referencia relacionada

Información de configuración de ejemplo

Envío y recepción de mensajes

La figura siguiente muestra el modelo de gestión de colas distribuidas, que detalla las relaciones entre entidades cuando se transmiten mensajes. También muestra el flujo de control.

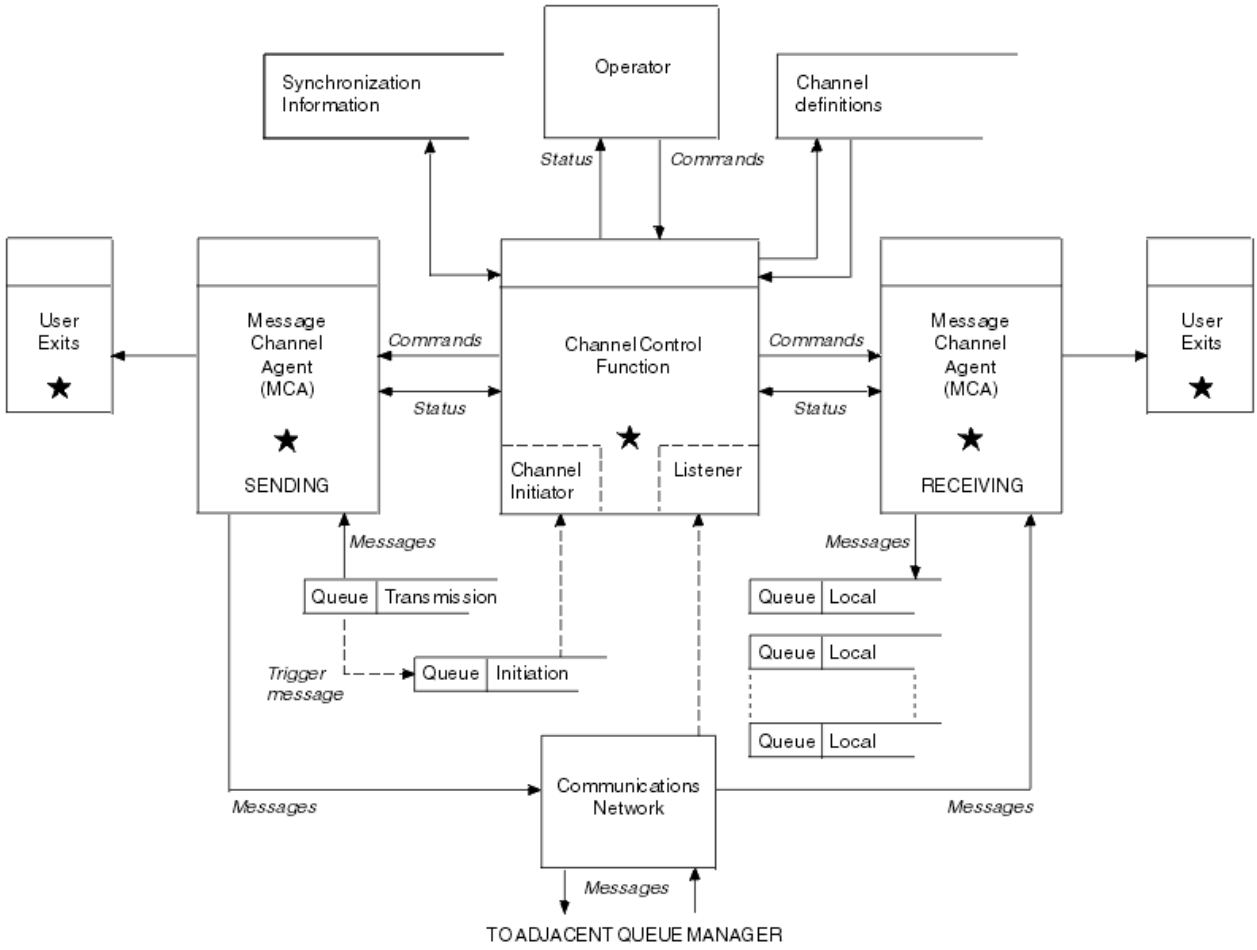


Figura 15. modelo de gestión de colas distribuidas

Nota:

1. Sólo hay un MCA por canal, en función de la plataforma. Puede haber una o varias funciones de control de canales para un gestor de colas determinado.
2. La implementación de los MCA y de las funciones de control de canales depende en gran medida de la plataforma. Pueden ser programas o procesos o hebras y pueden ser una sola entidad o muchas que comprenden varias partes enlazadas o independientes.
3. Todos los componentes marcados con una estrella pueden utilizar la MQI.

Parámetros de canal

Un MCA recibe sus parámetros en una de varias maneras:

- Si se ha iniciado mediante un mandato, se pasa un área de datos al nombre del canal. A continuación, el MCA lee la definición de canal directamente para obtener sus atributos.
- Para el remitente, y en algunos casos los canales de servidor, el MCA puede iniciarse automáticamente por el desencadenante del gestor de colas. El nombre del canal se recupera de la definición de proceso desencadenante, donde sea aplicable y pasa al MCA. El proceso restante es el mismo que se ha descrito anteriormente. Los canales de servidor sólo deben configurarse para desencadenar si están completos, es decir, si especifican un CONNAME al que conectarse.
- Si se ha iniciado de forma remota mediante un emisor, un servidor, un peticionario o de conexión con el cliente, se pasa al nombre del canal los datos iniciales del agente de canal de mensajes asociado. El MCA lee la definición de canal directamente para obtener sus atributos.

Algunos atributos no definidos en la definición de canal también son negociables:

Mensajes de división
Si un extremo no da soporte a los mensajes de división entonces los mensajes de división no se envían.

Capacidad de conversión
Si un extremo no puede realizar la conversión de la página de códigos necesaria o la conversión de codificación numérica cuando sea necesario, el otro extremo debe gestionarla. Si ningún extremo la soporte, cuando sea necesario, el canal no se puede iniciar.

Soporte de lista de distribución
Si un extremo no da soporte a listas de distribución, el MCA asociado establece un distintivo en su cola de transmisión de modo que sepa interceptar mensajes dirigidos a varios destinos.

Estado del canal y números de secuencia

Los programas de agente de canal de mensajes mantienen registros del número de secuencia actual y del número de unidad lógica de trabajo para cada canal, así como del estado general del canal. Algunas plataformas le permiten visualizar esta información de estado para ayudarle a controlar canales.

Cómo enviar un mensaje a otro gestor de colas

En este apartado se describe la forma más sencilla de enviar un mensaje entre gestores de colas, incluidos los requisitos previos y las autorizaciones necesarias. También se pueden utilizar otros métodos para enviar mensajes a un gestor de colas remoto.

Antes de enviar un mensaje de un gestor de colas a otro, deberá realizar los pasos siguientes:

1. Compruebe que el protocolo de comunicación elegido está disponible.
2. Inicie los gestores de colas.
3. Inicie los iniciadores de canal.
4. Inicie los escuchas.

También necesita tener la autorización de seguridad de IBM MQ correcta para crear los objetos necesarios.

Para enviar mensajes desde un gestor de colas a otro:

- Defina los objetos siguientes en el gestor de colas de origen:
  - Canal emisor
  - Definición de cola remota
  - Cola de inicio (z/OS necesaria en z/OS, de lo contrario es opcional)
  - Cola de transmisión
  - Cola de mensajes no entregados
- Defina los objetos siguientes en el gestor de colas de destino:
  - Canal receptor
  - Cola de destino
  - Cola de mensajes no entregados

Puede utilizar varios métodos diferentes para definir estos objetos, en función de la plataforma de IBM MQ:

- En todas las plataformas, puede utilizar los mandatos de script de IBM MQ (MQSC) descritos en Los mandatos MQSC, los mandatos de formato de mandato programable (PCF) descritos en Automatización de tareas de administración, o IBM MQ Explorer.
- z/OS En z/OS, también puede utilizar los paneles de operaciones y los paneles de control descritos en Administración de IBM MQ for z/OS .

- IBM i En IBM i, también puede utilizar la interfaz de panel.

Consulte los subtemas siguientes para obtener más información sobre la creación de componentes para enviar mensajes a otro gestor de colas:

Conceptos relacionados

“Técnicas de gestión de colas distribuidas de IBM MQ” en la página 211
Los subtemas de esta sección describen técnicas que se deben utilizar cuando planifique canales. Estos subtemas describen técnicas que le permitirán planificar la forma de conectar los gestores de colas y gestionar el flujo de mensajes entre las aplicaciones.

“Introducción a la gestión de colas distribuidas” en la página 231
La gestión de colas distribuidas (DQM) se utiliza para definir y controlar la comunicación entre los gestores de colas.

“Desencadenamiento de canales” en la página 255
IBM MQ proporciona un recurso para iniciar una aplicación automáticamente cuando se cumplen ciertas condiciones en una cola. Este recurso se denomina desencadenamiento.

“Seguridad de mensajes” en la página 253
Además de las funciones de recuperación habituales de IBM MQ, la gestión de colas distribuidas garantiza la entrega correcta de los mensajes utilizando un procedimiento de punto de sincronización coordinado entre los dos extremos del canal de mensajes. Si este procedimiento detecta un error, cierra el canal para que pueda investigar el problema y mantiene los mensajes de forma segura en la cola de transmisión hasta que se reinicia el canal.

Tareas relacionadas

“Creación de gestores de colas en Multiplatforms” en la página 7
Antes de poder utilizar mensajes y colas, debe crear e iniciar al menos un gestor de colas y los objetos asociados al mismo. Un gestor de colas gestiona los recursos que tiene asociados, en particular las colas que posee. Proporciona servicios de colocación en cola a las aplicaciones para llamadas y mandatos MQI (Message Queuing Interface) para crear, modificar, mostrar y suprimir objetos de IBM MQ.

“Supervisión y control de canales en AIX, Linux, and Windows” en la página 263
Para DQM debe crear, supervisar y controlar los canales con los gestores de colas remotos. Puede controlar los canales utilizando mandatos, programas, IBM MQ Explorer, archivos para las definiciones de canal y un área de almacenamiento para la información de sincronización.

“Supervisión y control de canales en IBM i” en la página 287
Utilice los mandatos y paneles de DQM para crear, supervisar y controlar los canales con gestores de colas remotos. Cada gestor de colas tiene un programa DQM para controlar las interconexiones con gestores de colas remotos compatibles.

“Configuración de conexiones entre el cliente y el servidor” en la página 16
Para configurar los enlaces de comunicación entre IBM MQ MQI clients y servidores, decida el protocolo de comunicación, defina las conexiones en ambos extremos del enlace, inicie un escucha y defina canales.

“Configuración de un clúster de gestores de colas” en la página 309
Los clústeres proporcionan un mecanismo para interconectar gestores de colas de forma que simplifica la configuración inicial y la gestión continua. Puede definir componentes de clúster, y crear y gestionar los clústeres.

“Setting up communications with other queue managers on z/OS” en la página 975
This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

Definir los canales

Para enviar mensajes de un gestor de colas a otro, debe definir dos canales. Debe definir un canal en el gestor de colas de origen y un canal en el gestor de colas de destino.

En el gestor de colas de origen

Defina un canal con un tipo de canal SENDER (emisor). Debe especificar lo siguiente:

- El nombre de la cola de transmisión que se va a utilizar (atributo XMITQ).
- El nombre de conexión del sistema asociado (atributo CONNAME).
- El nombre del protocolo de comunicaciones que esté utilizando (atributo TRPTYPE). En IBM MQ for z/OS, el protocolo debe ser TCP o LU6.2. En Multiplatforms, no tiene que especificar el protocolo. Puede dejarlo para obtener el valor de la definición de canal predeterminada.

En la sección Atributos de canal se ofrecen detalles de todos los atributos de canal.

En el gestor de colas de destino

Defina un canal con un tipo de canal RECEIVER (receptor) y el mismo nombre que el canal emisor.

Especifique el nombre del protocolo de comunicaciones que esté utilizando (atributo TRPTYPE). En IBM MQ for z/OS, el protocolo debe ser TCP o LU6.2. En Multiplatforms, no tiene que especificar el protocolo. Puede dejarlo para obtener el valor de la definición de canal predeterminada.

Las definiciones de canal receptor pueden ser genéricas. Esto significa que si tiene varios gestores de colas comunicándose con el mismo receptor, todos los canales emisores pueden especificar el mismo nombre para el receptor y una definición de canal receptor se aplica a todos ellos.

Tras haber definido el canal, puede probarlo mediante el mandato PING CHANNEL. Este mandato envía un mensaje especial del canal emisor al canal receptor y comprueba que se devuelve.

Nota: El valor del parámetro TRPTYPE es ignorado por el agente de canal de mensajes correspondiente. Por ejemplo, un TRPTYPE de TCP en la definición de canal emisor se inicia con un TRPTYPE de LU62 en la definición de canal receptor como un socio.

Definición de las colas

Para enviar mensajes de un gestor de colas a otro, debe definir hasta seis colas. Debe definir hasta cuatro colas en el gestor de colas de origen y hasta dos colas en el gestor de colas de destino.

En el gestor de colas de origen

- Definición de cola remota

  En esta definición, especifique lo siguiente:

  Nombre del gestor de colas remoto
  Nombre del gestor de colas de destino.

  Nombre de cola remota
  Nombre de la cola de destino en el gestor de colas de destino.

  Nombre de cola de transmisión
  Nombre de la cola de transmisión. No es necesario especificar este nombre de cola de transmisión. De lo contrario, se utiliza una cola de transmisión con el mismo nombre que el gestor de colas de destino. Si no existe, se utiliza la cola de transmisión predeterminada. Se recomienda dar a la cola de transmisión el mismo nombre que el gestor de colas de destino para que la cola se encuentre de forma predeterminada.

- Definición de cola de inicialización

  z/OS Es obligatorio. Debe utilizar la cola de inicio denominada SYSTEM.CHANNEL.INITQ.

  Multi Es opcional. Considere la posibilidad de denominar la cola de inicio SYSTEM.CHANNEL.INITQ.

- Definición de cola de transmisión

  Una cola local con el atributo USAGE establecido en XMITQ. IBM i Si está utilizando la interfaz nativa de IBM MQ for IBM i, el atributo USAGE es *TMQ.

- Definición de cola de mensajes no entregados

  Defina una cola de mensajes no entregados en la que se pueden escribir mensajes no entregados.

En el gestor de colas de destino

- Definición de cola local

  La cola de destino. El nombre de esta cola debe ser el mismo que el especificado en el campo de nombre de cola remota de la definición de cola remota en el gestor de colas de origen.

- Definición de cola de mensajes no entregados

  Defina una cola de mensajes no entregados en la que se pueden escribir mensajes no entregados.

Conceptos relacionados

"Crear una cola de transmisión" en la página 237
Antes de poder iniciar un canal (distinto del canal peticionario), la cola de transmisión debe definirse como se describe en esta sección. Se debe asignar un nombre a la cola de transmisión en la definición de canal.

"Creación de una cola de transmisión en IBM i" en la página 237
Puede crear una cola de transmisión en la plataforma de IBM i utilizando el panel Crear cola MQM.

Crear una cola de transmisión

Antes de poder iniciar un canal (distinto del canal peticionario), la cola de transmisión debe definirse como se describe en esta sección. Se debe asignar un nombre a la cola de transmisión en la definición de canal.

Defina una cola local con el atributo USAGE establecido en XMITQ para cada canal emisor de mensajes. Si desea utilizar una cola de transmisión específica en las definiciones de colas remotas, cree una cola remota como se muestra a continuación.

Para crear una cola de transmisión, utilice los mandatos de IBM MQ (MQSC), como se muestra en los ejemplos siguientes:

Ejemplo de creación de cola de transmisión

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') USAGE(XMITQ)
```

Ejemplo de creación de cola remota

```
DEFINE QREMOTE(PAYROLL) DESCR('Remote queue for QM2') +
XMITQ(QM2) RNAME(PAYROLL) RQMNAME(QM2)
```

Considere la posibilidad de asignar un nombre a la cola de transmisión con el nombre del gestor de colas del sistema remoto, como se muestra en los ejemplos.

Creación de una cola de transmisión en IBM i

Puede crear una cola de transmisión en la plataforma de IBM i utilizando el panel Crear cola MQM.

Debe definir una cola local con el atributo de campo Uso establecido en TMQ, para cada canal de mensajes emisores.

Si desea utilizar definiciones de colas remotas, utilice el mismo mandato para crear una cola de tipo *RMT y el uso de *NORMAL.

Para crear una cola de transmisión, utilice el mandato CRTMQMQ desde la línea de mandatos para que aparezca el primer panel de creación de colas; consulte la Figura 16 en la página 238.

```
Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

Queue name . . . . . .

Queue type . . . . . .  ____     *ALS, *LCL, *MDL, *RMT

Message Queue Manager name . . .  *DFT_____________________________
_____









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

Figura 16. Crear una cola (1)

Escriba el nombre de la cola y especifique el tipo de cola que desea crear: Local, Remota o Alias. Para una cola de transmisión, especifique Local (*LCL) en este panel y pulse Intro.

Se le presentará la segunda página del panel Crear cola MQM; consulte la Figura 17 en la página 238.

```
Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

Queue name . . . . . > HURS.2.HURS.PRIORIT

Queue type . . . . . > *LCL     *ALS, *LCL, *MDL, *RMT
Message Queue Manager name . . .  *DFT
Replace . . . . . .  *NO      *NO, *YES
Text 'description' . . . . . .  '                  '
Put enabled . . . . . .  *YES     *SYSDFTQ, *NO, *YES
Default message priority . . . .  0       0-9, *SYSDFTQ
Default message persistence . .  *NO      *SYSDFTQ, *NO, *YES
Process name . . . . . .  '                  '
Triggering enabled . . . . . .  *NO      *SYSDFTQ, *NO, *YES
Get enabled . . . . . .  *YES     *SYSDFTQ, *NO, *YES
Sharing enabled . . . . . .  *YES     *SYSDFTQ, *NO, *YES



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

Figura 17. Crear una cola (2)

Cambie cualquiera de los valores predeterminados mostrados. Pulse Av Pág para ir a la siguiente pantalla; consulte la Figura 18 en la página 239.

```
Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

Default share option . . . . . *YES     *SYSDFTQ, *NO, *YES
Message delivery sequence . . . *PTY     *SYSDFTQ, *PTY, *FIFO
Harden backout count . . . . . *NO      *SYSDFTQ, *NO, *YES
Trigger type . . . . . *FIRST    *SYSDFTQ, *FIRST, *ALL...
Trigger depth . . . . . 1       1-999999999, *SYSDFTQ
Trigger message priority . . . . 0       0-9, *SYSDFTQ
Trigger data . . . . . '                 '
Retention interval . . . . . 999999999   0-999999999, *SYSDFTQ
Maximum queue depth . . . . . 5000     1-24000, *SYSDFTQ
Maximum message length . . . . . 4194304    0-4194304, *SYSDFTQ
Backout threshold . . . . . 0       0-999999999, *SYSDFTQ
Backout requeue queue . . . . . '                 '
Initiation queue . . . . . '                 '



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

Figura 18. Crear una cola (3)

Escriba *TMQ, para la cola de transmisión, en el campo Uso de este panel y cambie cualquiera de los valores predeterminados que aparecen en los demás campos.

```
Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

Usage . . . . . *TMQ     *SYSDFTQ, *NORMAL, *TMQ
Queue depth high threshold . . . 80      0-100, *SYSDFTQ
Queue depth low threshold . . . 20      0-100, *SYSDFTQ
Queue full events enabled . . . *YES     *SYSDFTQ, *NO, *YES
Queue high events enabled . . . *YES     *SYSDFTQ, *NO, *YES
Queue low events enabled . . . . *YES     *SYSDFTQ, *NO, *YES
Service interval . . . . . 999999999   0-999999999, *SYSDFTQ
Service interval events . . . . *NONE     *SYSDFTQ, *HIGH, *OK, *NONE
Distribution list support . . . *NO      *SYSDFTQ, *NO, *YES
Cluster Name . . . . . *SYSDFTQ
Cluster Name List . . . . . *SYSDFTQ
Default Binding . . . . . *SYSDFTQ   *SYSDFTQ, *OPEN, *NOTFIXED




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

Figura 19. Crear una cola (4)

Cuando esté convencido de que los campos contienen los datos correctos, pulse Intro para crear la cola.

Inicio del canal

Al colocar mensajes en la cola remota definida en el gestor de colas de origen, se almacenan en la cola de transmisión hasta que se inicia el canal. Cuando el canal se ha iniciado, los mensajes se entregan a la cola de destino en el gestor de colas remoto.

Inicie el canal en el gestor de colas emisor mediante el mandato START CHANNEL. Cuando se inicie el canal emisor, el canal receptor se inicia automáticamente (por el escucha) y los mensajes se envían a la cola de destino. Ambos extremos del canal de mensajes deben estar en ejecución para que se transfieran los mensajes.

Dado que los dos extremos del canal están en diferentes gestores de colas, podrían haberse definido con diferentes atributos. Para resolver las diferencias, existe una negociación inicial de datos entre los dos extremos cuando se inicia el canal. En general, los dos extremos del canal funcionan con atributos que requieren menos recursos. Esto permite que los grandes sistemas alberguen la menor cantidad de recursos de los sistemas más pequeños en el otro extremo del canal de mensajes.

El agente de canal de mensajes (MCA) emisor divide los grandes mensajes antes de enviarlos por el canal. Se reagrupan en el gestor de colas remoto. Esto no es evidente para el usuario.

Un agente de canal de mensajes (MCA) puede transferir mensajes utilizando varias hebras. Este proceso, denominado *canalización* permite que el MCA transfiera los mensajes de forma más eficaz, con menos estados de espera. El proceso de canalización mejora el rendimiento del canal.

Función de control de canales

La función de control de canales proporciona recursos para definir, supervisar y controlar canales.

Los mandatos se emiten a través de paneles, programas o desde una línea de mandatos para la función de control de canales. La interfaz de panel también muestra el estado del canal y los datos de definición de canal. Puede utilizar mandatos PCF (formato de mandato programable) o los mandatos de IBM MQ (MQSC) y los mandatos de control que se detallan en "Supervisión y control de canales en AIX, Linux, and Windows" en la página 263.

Los mandatos se clasifican en los grupos siguientes:

- Administración de canales
- Control de canales
- Supervisión del estado del canal

Los mandatos de administración de canales se ocupan de las definiciones de los canales. Le permiten:

- Crear una definición de canal
- Copiar una definición de canal
- Modificar una definición de canal
- Suprimir una definición de canal

Los mandatos de control de canales gestionan el funcionamiento de los canales. Le permiten:

- Iniciar un canal
- Detener un canal
- Volver a sincronizar con la aplicación asociada (en algunas implementaciones)
- Restablecer los números de secuencia de los mensajes
- Resolver un lote pendiente de mensajes
- Ejecutar un mandato ping; enviar una comunicación de prueba a través del canal

La supervisión de canales muestra el estado de los canales, por ejemplo:

- Valores actuales del canal
- Si el canal está activo o inactivo
- Si el canal ha terminado en un estado sincronizado

Conceptos relacionados

Dónde encontrar información para ayudar con la determinación de problemas

Preparación de canales

Antes de intentar iniciar un canal de mensajes o un canal MQI, debe preparar el canal. Debe asegurarse de que todos los atributos de las definiciones de canal local y remoto son correctos y compatibles.

La sección Atributos de canal describe las definiciones y los atributos de canal.

Aunque configure definiciones de canal explícitas, las negociaciones de canal realizadas cuando se inicia un canal pueden alterar temporalmente uno u otro de los valores definidos. Este comportamiento es normal y no es evidente para el usuario y se ha organizado de este modo para que las definiciones incompatibles puedan trabajar conjuntamente.

Definición automática de canales de conexión con el receptor y el servidor.

En IBM MQ en Multiplatforms, si no hay una definición de canal adecuada, para un canal receptor o de conexión de servidor que tenga habilitada la definición automática, se crea automáticamente una definición. La definición se crea utilizando:

1. La definición de canal modelo adecuada, SYSTEM.AUTO.RECEIVER o SYSTEM.AUTO.SVRCONN. Las definiciones de canal modelo para la definición automática son las mismas que los valores predeterminados del sistema, SYSTEM.DEF.RECEIVER y SYSTEM.DEF.SVRCONN, excepto para el campo de descripción, que es "Definido automáticamente por" seguido de 49 espacios en blanco. El administrador de sistemas puede optar por cambiar cualquier parte de las definiciones de canal modelo suministradas.
2. Información del sistema socio. Los valores del socio se utilizan para el nombre de canal y el valor de reinicio de número de secuencia.
3. Un programa de salida de canal, que puede utilizar para modificar los valores creados por la definición automática. Consulte Programa de salida de definición automática de canal.

A continuación, la descripción se comprueba para determinar si se ha alterado por una salida de definición automática o porque la definición de modelo ha cambiado. Si los primeros 44 caracteres todavía son "Definido automáticamente por" seguidos de 29 blancos, se añade el nombre del gestor de colas. Si los 20 caracteres finales siguen estando todos el blanco, se añaden la hora y la fecha local.

Cuando se ha creado la definición y se ha almacenado el canal, el inicio prosigue como si la definición hubiera existido siempre. El tamaño del lote, el tamaño de la transmisión y el tamaño del mensaje se negocian con el socio.

Definición de otros objetos

Antes de que se pueda iniciar un canal de mensajes, deben definirse ambos extremos (o habilitarse para la definición automática) en los gestores de colas. La cola de transmisión a la que ha de servir debe estar definida en el gestor de colas en el extremo emisor. El enlace de comunicaciones debe estar definido y disponible. Podría ser necesario que prepare otros objetos de IBM MQ, tales como definiciones de colas remotas, definiciones de alias de gestor de colas y definiciones de alias de colas de respuesta, para implementar los escenarios que se describen en “Configuración de la gestión de colas distribuidas” en la página 210.

Para obtener información sobre la definición de canales MQI, consulte “Definición de canales MQI” en la página 31.

Varios canales de mensajes por cola de transmisión

Es posible definir más de un canal por cola de transmisión, pero sólo uno de estos canales puede estar activo en cualquier momento. Tenga en cuenta esta opción para la prestación de rutas alternativas entre gestores de colas para equilibrar el tráfico y realizar acciones correctivas en anomalías de enlace. Una cola de transmisión no puede ser utilizada por otro canal si el canal anterior para utilizarla acabó dejando un lote de mensajes pendiente en el extremo emisor. Para obtener más información, consulte “Manejo de canales pendientes” en la página 252.

Iniciar un canal

Se puede hacer que un canal empiece a transmitir mensajes de una de cuatro maneras: Puede:

- Iniciarse por un operador (que no sea un canal receptor, de clúster receptor o de conexión del servidor).
- Desencadenarse desde la cola de transmisión. Este método se aplica a canales emisores y canales de servidor totalmente calificado (aquellos canales que especifican un CONNAME) solamente. Debe preparar los objetos necesarios para canales desencadenantes.
- Iniciarse desde un programa de aplicación (que no sea un canal receptor, de clúster receptor o de conexión con el servidor).
- Iniciarse de forma remota desde la red por un canal emisor, de clúster emisor, peticionario, de servidor o de conexión con el cliente. Las transmisiones de canal receptor, de clúster receptor y peticionario se inician de este modo; al igual que los canales de conexión con el servidor. Los propios canales ya deben estar iniciados (es decir habilitados).

Nota: Que un canal esté 'iniciado' no significa necesariamente que esté transmitiendo mensajes. En cambio, puede estar 'habilitado' para iniciar la transmisión cuando se produce uno de los cuatro sucesos anteriormente descritos. La habilitación e inhabilitación de un canal se logra mediante los mandatos del operador START y STOP.

Estados de un canal

Un canal puede estar en cualquier momento en uno de los muchos estados que existen. Algunos estados también tienen subestados. A partir de un estado determinado un canal puede pasar a otros estados.

En la Figura 20 en la página 242 se muestra la jerarquía de todos los estados de canal posibles y los subestados aplicables a cada uno de los estados de canal.

En la Figura 21 en la página 243 se muestran los enlaces entre estados de canal. Estos enlaces se aplican a todos los tipos de canal de mensajes y canales de conexión de servidor.

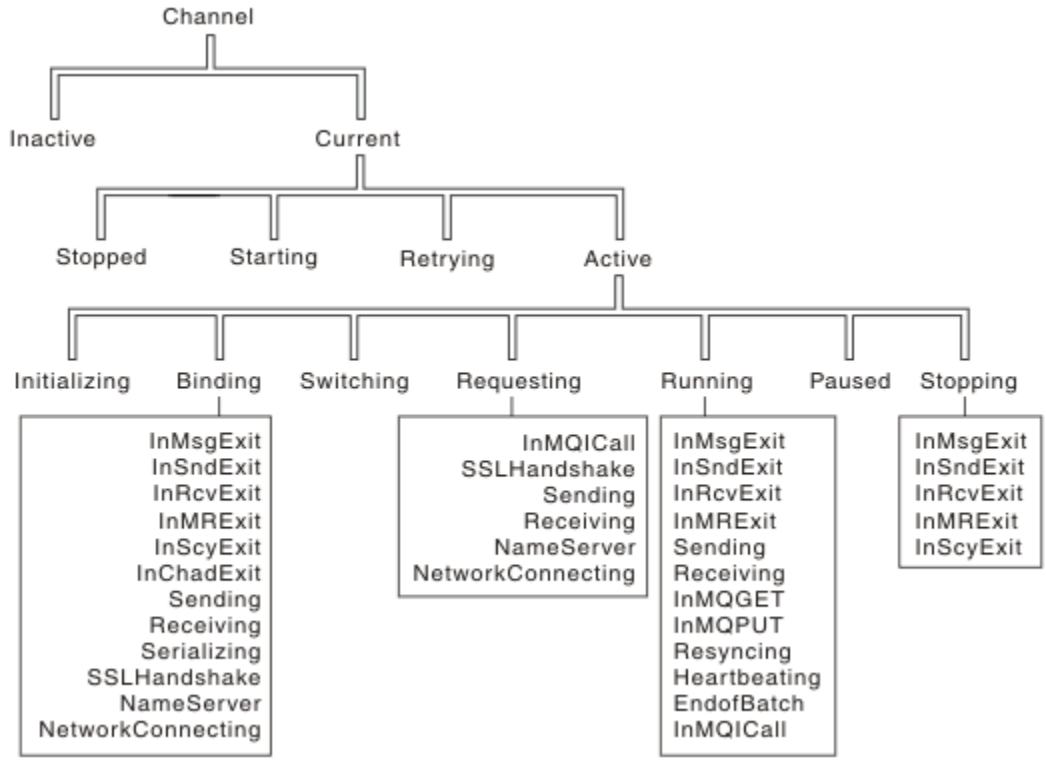


Figura 20. Estados y subestados de un canal

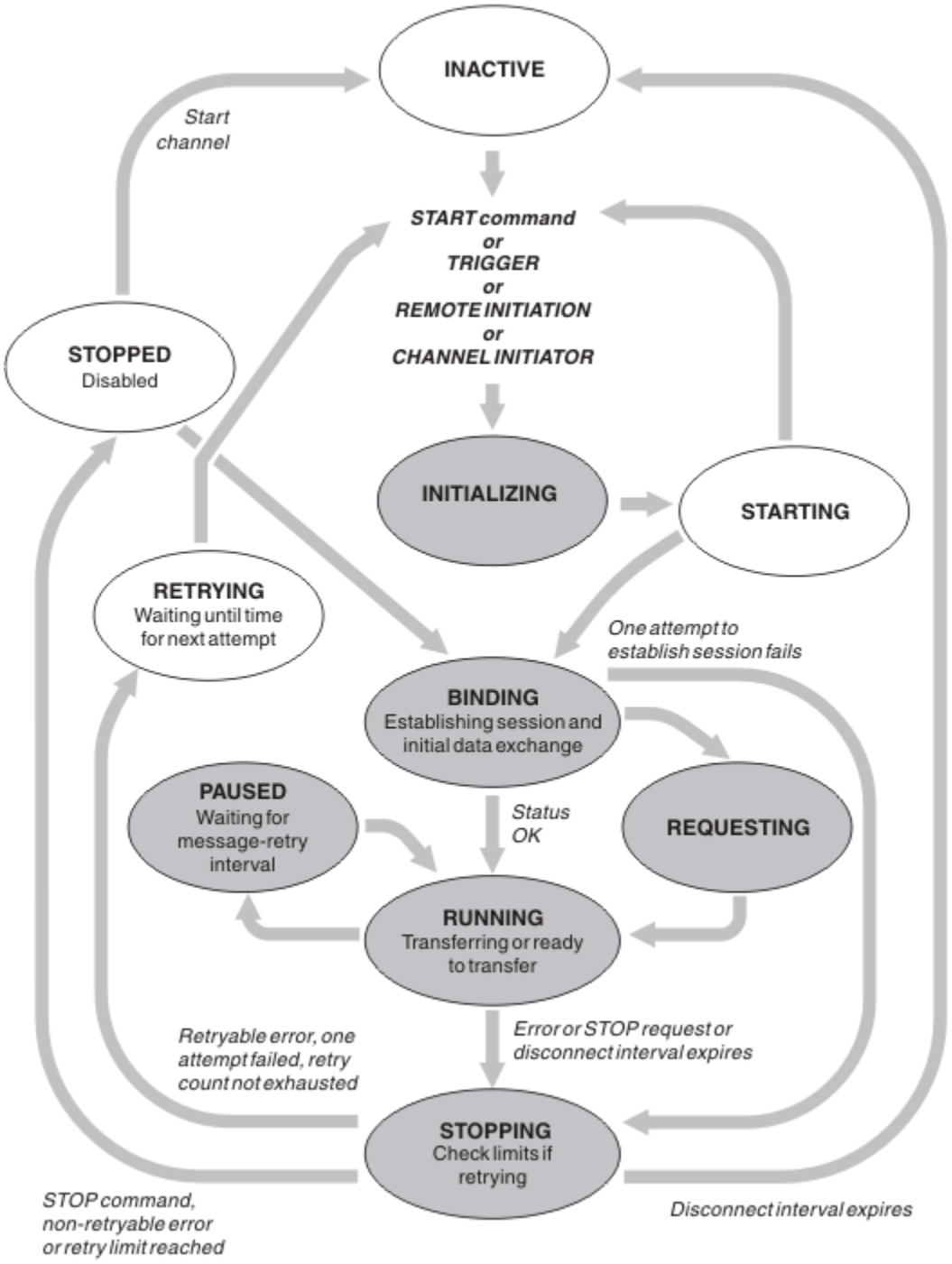


Figura 21. Flujos entre estados de canal

Actual y activo

Un canal es el canal *actual* si se encuentra en cualquier estado excepto el estado inactivo. Un canal actual está *activo* a menos que esté en los estados REINTENTANDO, DETENIDO o INICIANDO. Cuando un canal está activo, consume recursos y se ejecuta un proceso o hebra. Los siete estados posibles de un canal activo (INITIALIZING, BINDING, SWITCHING, REQUESTING, RUNNING, PAUSED o STOPPING) aparecen resaltados en la Figura 21 en la página 243.

Un canal activo también puede mostrar un subestado que ofrecerá más detalles sobre lo que está haciendo exactamente el canal. Los subestados de cada estado se muestran en la Figura 20 en la página 242.

Actual y activo
El canal es "actual" si se encuentra en cualquier otro estado que no sea inactivo. Un canal actual está "activo" a menos que esté en los estados REINTENTANDO, DETENIDO o INICIANDO.

Si un canal está "activo" también puede mostrar un subestado que proporciona más detalles de lo que el canal está haciendo exactamente.

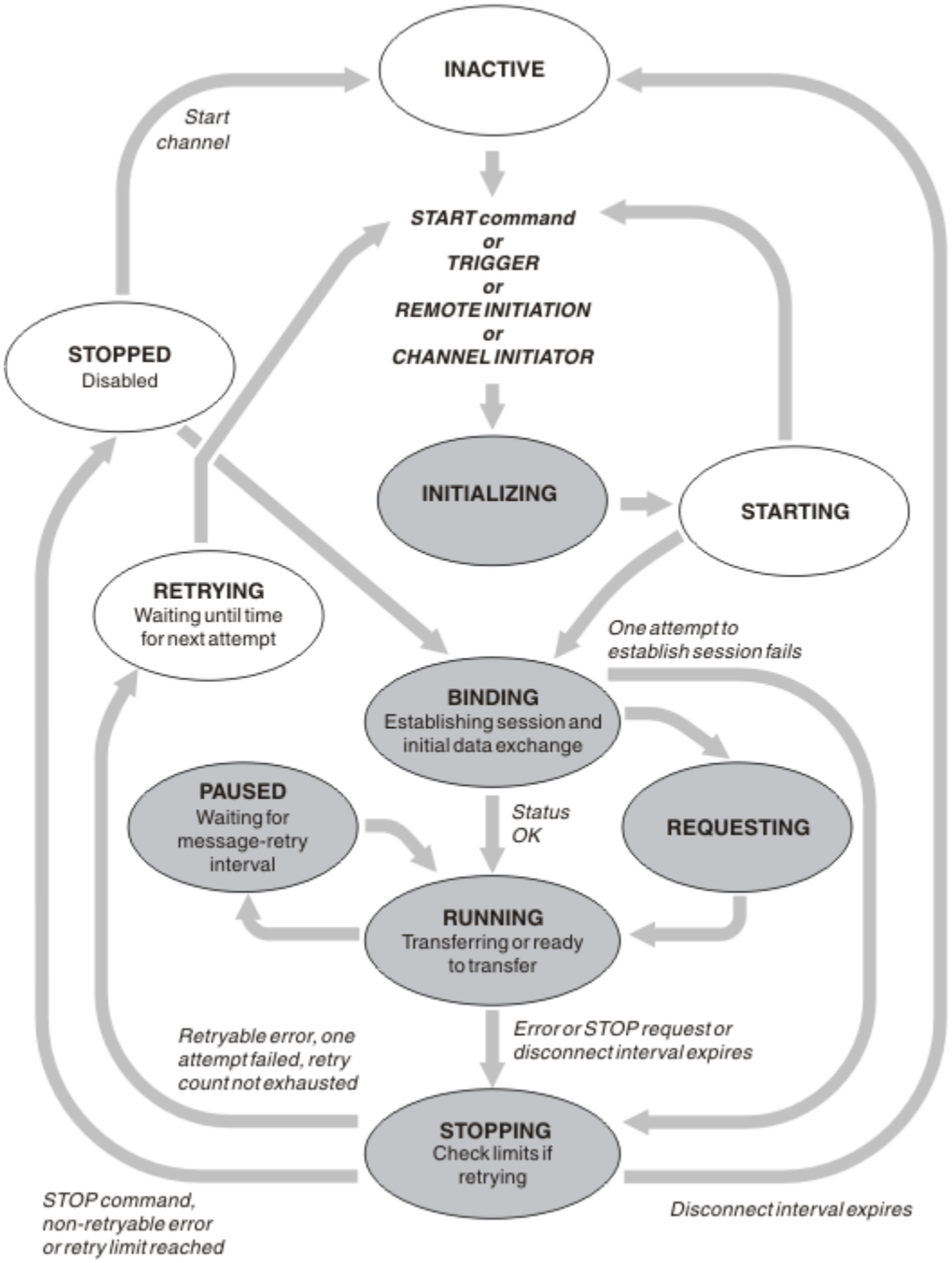


Figura 22. Flujos entre estados de canal

Nota:

1. Cuando un canal está en uno de los seis estados resaltados en la Figura 22 en la página 244 (INITIALIZING, BINDING, REQUESTING, RUNNING, PAUSED o STOPPING), está consumiendo recursos y un proceso o hebra está en ejecución; el canal está *activo*.

2. Cuando un canal está en estado STOPPED, la sesión puede estar activa debido a que el estado siguiente todavía no se conoce.

Especificación del número máximo de canales actuales

Puede especificar el número máximo de canales que pueden ser actuales simultáneamente. Este número es el número de canales que tienen entradas en la tabla de estados de canal, incluidos los canales que se están reintentando y los canales que están detenidos. Especifíquelo para su plataforma:

- z/OS Utilice el mandato `ALTER QMGR MAXCHL`.
- IBM i Edite el archivo de inicialización del gestor de colas.
- Linux AIX Edite el archivo de configuración del gestor de colas.
- Utilizar IBM MQ Explorer.

Para obtener más información sobre los valores que se pueden establecer utilizando el archivo de configuración o inicialización, consulte Stanzas del archivo de configuración para la gestión de colas distribuidas. Para obtener más información sobre cómo especificar el número máximo de canales, consulte los temas siguientes:

- ALW Administración de IBM MQ.
- IBM i Administración de IBM MQ for IBM i.
- z/OS Administración de IBM MQ for z/OS.

Nota:

1. Los canales de conexión con el servidor se incluyen en este número.
2. Un canal debe ser actual antes de que pueda ser activo. Si un canal se ha iniciado, pero no puede convertirse en el actual, el inicio falla.

Especificación del número máximo de canales activos

También puede especificar el número máximo de canales activos para evitar que el sistema se sobrecargue con muchos canales de inicio. Si utiliza este método, establezca el atributo de intervalo de desconexión en un valor bajo para permitir que los canales en espera se inicien en cuanto terminen los otros canales.

Cada vez que un canal que está reintentando intenta establecer conexión con su socio, debe convertirse en un canal activo. Si el intento falla, sigue siendo un canal actual que no está activo, hasta que es hora del siguiente intento. El número de veces que un canal realiza reintentos y con qué frecuencia, viene determinado por los atributos de número de reintentos y de intervalo de reintento de canal. Existen valores cortos y largos para estos dos atributos. Consulte Atributos de canal para obtener más información.

Cuando un canal tiene que convertirse en un canal activo (debido a que se ha emitido un mandato START, o porque se ha desencadenado, o porque es hora de otro reintento), pero no ha podido hacerlo porque el número de canales activos ya está en el valor máximo, el canal espera hasta que una de las ranuras activas sea liberada por otra instancia de canal que deja de ser activa. Sin embargo, si, un canal se inicia porque se inicia de forma remota y no hay ranuras activas disponibles en ese momento, el inicio remoto se rechaza.

Cada vez que un canal, que no sea un canal peticionario, está intentando ser activo, entra en el estado STARTING. Este estado se produce incluso si existe una ranura activa inmediatamente disponible, aunque es sólo en el estado STARTING durante un breve tiempo. Sin embargo, si el canal debe esperar una ranura activa, está en estado STARTING mientras espera.

Los canales peticionarios no entran en estado STARTING. Si un canal peticionario no puede iniciar debido a que el número de canales activos ya se encuentra en el límite, el canal finaliza de forma anómala.

Cuando un canal, que no sea un canal peticionario, no puede obtener una ranura activa, y por tanto espera una, se graba un mensaje en el registro z/OS o la consola de z/OSe, y se genera un suceso. Cuando más tarde se libera una ranura y el canal no puede adquirirla, se generan otro mensaje y otro suceso. No se genera ninguno de estos sucesos y mensajes si el canal puede adquirir una ranura inmediatamente.

Si se emite un mandato STOP CHANNEL mientras el canal está esperando activarse, el canal entra en el estado STOPPED. Se genera un suceso detenido por el canal.

Los canales de conexión con el servidor están incluidos en el número máximo de canales activos.

Para obtener más información sobre cómo especificar el número máximo de canales activos, consulte los temas siguientes:

- ALW Administración de IBM MQ.
- IBM i Administración de IBM MQ for IBM i.
- z/OS Administración de IBM MQ for z/OS.

Errores de canal
Los errores en los canales hacen que el canal deje de realizar transmisiones. Si el canal es un emisor o servidor, entra en el estado RETRY porque es posible que el problema se resuelva por si solo. Si no puede entrar en el estado RETRY, el canal entra en el estado STOPPED.

Para los canales emisores, la cola de transmisión asociada se establece en GET(DISABLED) y el desencadenamiento está desactivado. (Un mandato STOP con STATUS(STOPPED) lleva el lado que lo emitió al estado STOPPED; sólo la caducidad del intervalo de desconexión o un mandato STOP con STATUS(INACTIVE) hace que finalice normalmente y quede inactivo.) Los canales cuyo estado es STOPPED necesitan la intervención del operador para poder reiniciarse (consulte “Reinicio de canales detenidos” en la página 251).

Nota: Para sistemas IBM i IBM i, AIX, Linux, and Windows, se debe estar ejecutando un iniciador de canal para que se pruebe el reintento. Si el iniciador de canal no está disponible, el canal pasa a estar inactivo y se debe reiniciar manualmente. Si utiliza un script para iniciar el canal, asegúrese de que se ejecuta el iniciador de canal antes de intentar ejecutar el script.

Cuenta de reintentos largos (LONGRTY) describe cómo funciona el reintento. Si el error se borra, el canal se reinicia automáticamente y la cola de transmisión se rehabilita. Si se alcanza el límite de reintentos sin que se borre el error, el canal pasa al estado STOPPED. Un canal detenido debe reiniciarse manualmente por el operador. Si el error sigue presente, no vuelva a intentarlo. Cuando se inicia satisfactoriamente, la cola de transmisión se rehabilita.

z/OS Si el iniciador de canal se detiene mientras un canal está en estado RETRYING o STOPPED, el estado de canal se recuerda cuando se reinicia el iniciador de canal. Sin embargo, el estado del canal para el tipo de canal SVRCONN se restablece si el iniciador de canal se detiene mientras el canal está en estado STOPPED.

Multi Si el gestor de colas se detiene mientras un canal está en estado RETRYING o STOPPED, el estado de canal se recuerda cuando se reinicia el gestor de colas. A partir de IBM MQ 8.0, estos se aplica también a los canales SVRCONN. Anteriormente, el estado de canal para el tipo de canal SVRCONN se ha restablecido si el iniciador de canal se ha detenido mientras el canal estaba en estado STOPPED.

Si un canal es incapaz de transferir un mensaje a la cola de destino porque dicha cola está llena o put inhibido, el canal puede reintentar la operación un número de veces (especificado en el atributo de número de reintentos de mensaje) a un intervalo determinado (especificado en el atributo de intervalo de reintentos de mensaje). Como alternativa, puede escribir su propio programa de salida de reintento de mensaje que determina qué circunstancias causan un reintento y el número de intentos realizados. El canal entra en el estado PAUSED mientras espera que el intervalo de reintento de mensaje finalice.

Consulte la sección Atributos de canal para obtener información sobre los atributos de canal y la sección Programas de salida de canal para canales de mensajería para obtener información sobre la salida de reintento de mensaje.

Límites de canal de conexión con el servidor

Puede establecer límites de canal de conexión de servidor para evitar que las aplicaciones cliente agoten los recursos de canal de gestor de colas con el parámetro **MAXINST** y para impedir que una única aplicación cliente agote la capacidad de canal de conexión de servidor con el parámetro **MAXINSTC** .

Establezca **MAXINST** y **MAXINSTC** con el mandato **DEFINE CHANNEL** .

Un número máximo total de canales que pueden estar activos en cualquier momento en un solo gestor de colas. El número total de instancias de canal de conexión con el servidor se incluye en el número máximo de canales activos.

Si no especifica el número máximo de instancias simultáneas de un canal de conexión con el servidor que se pueden iniciar, es posible que una sola aplicación cliente que se conecte a un único canal de conexión con el servidor agote el número máximo de canales activos disponibles. Cuando se alcanza el número máximo de canales activos, ello impide que se inicien otros canales en el gestor de colas. Para evitar esta situación, debe limitar el número de instancias simultáneas de un canal específico de conexión con el servidor que se pueden iniciar, independientemente del cliente que las haya iniciado.

Si el valor del límite se reduce por debajo del número de instancias del canal de conexión con el servidor actualmente en ejecución, incluso a cero, los canales en ejecución no se ven afectados. No se podrán iniciar nuevas instancias hasta que haya dejado de ejecutarse un número suficiente de instancias existentes, de modo que el número de instancias actualmente en ejecución sea menor que el valor del límite.

Además, muchos canales diferentes de conexión con el cliente pueden conectarse a un canal específico de conexión con el servidor. El límite en el número de instancias simultáneas de un canal específico de conexión con el servidor que se pueden iniciar, independientemente del cliente que las haya iniciado, impide que un cliente agote la capacidad máxima de canales activos del gestor de colas. Si no limita también el número de instancias simultáneas de un canal específico de conexión con el servidor que se pueden iniciar desde un cliente determinado, es posible que una sola aplicación cliente anómala abra tantas conexiones que agote la capacidad del canal asignada a un solo canal de conexión con el servidor y, por lo tanto, impida que otros clientes que necesitan utilizar el canal se conecten con él. Para evitar esta situación, debe limitar el número de instancias simultáneas de un canal específico de conexión con el servidor que se pueden iniciar desde un solo cliente.

Si el valor del límite de clientes individuales se reduce por debajo del número de instancias del canal de conexión con el servidor actualmente en ejecución desde clientes individuales, incluso a cero, los canales en ejecución no se ven afectados. No obstante, no se podrán iniciar nuevas instancias del canal de conexión con el servidor desde un cliente individual que supere el nuevo límite hasta que haya dejado de ejecutarse un número suficiente de instancias existentes de dicho cliente, de modo que el número de instancias actualmente en ejecución sea menor que el valor de este parámetro.

Referencia relacionada

Atributos de canal y tipos de canal

DEFINE CHANNEL

Cómo comprobar que el otro extremo del canal sigue estando disponible

Puede utilizar el intervalo de pulsaciones, el intervalo de estado activo y el tiempo de espera de recepción, para comprobar que el otro extremo del canal está disponible.

Pulsaciones

Puede utilizar el atributo de canal Intervalo de pulsaciones para especificar que deben pasarse flujos desde el MCA emisor cuando no hay mensajes en la cola de transmisión, tal como se describe en Intervalo de pulsaciones (HBINT).

Mantener activo

z/OS En z/OS, si utiliza TCP/IP como protocolo de transporte, también puede especificar un valor para el atributo de canal de intervalo de **Keepalive** (**KAINT**). Se recomienda asignar al intervalo de **Keepalive** un valor mayor que el intervalo de pulsaciones y un valor menor que el valor de desconexión. Puede utilizar este atributo para especificar un valor de espera para cada canal, tal como se describe en Intervalo de estado activo (KAINT).

Multi En sistemas IBM i, AIX, Linux, and Windows , si utiliza TCP como protocolo de transporte, puede establecer `keepalive=yes`. Si especifica esta opción, TCP comprueba periódicamente si el otro extremo de la conexión sigue estando disponible. En caso contrario, el canal finaliza. Esta opción se describe en Intervalo de estado activo (KAINT).

Si tiene canales no fiables que sufren los errores de TCP, con el uso de la opción **Keepalive** los canales tienen más probabilidades de recuperación.

Puede especificar intervalos de tiempo para controlar el comportamiento de la opción **Keepalive**. Cuando se cambia el intervalo de tiempo, sólo los canales TCP/IP iniciados después del cambio se ven afectados. Asegúrese de que el valor que elige para el intervalo de tiempo sea inferior al valor del intervalo de desconexión para el canal.

Para obtener más información sobre cómo utilizar la opción **Keepalive** , consulte el parámetro **KAINT** en el mandato **DEFINE CHANNEL** .

Tiempo de espera de recepción

Si utiliza TCP como protocolo de transporte, el extremo de recepción de una conexión de canal no MQI desocupada también se cierra si no se reciben datos durante un período. Este periodo, el valor de *tiempo de espera de recepción* , se determina según el valor de **HBINT** (intervalo de pulsaciones).

En IBM MQ para sistemas IBM i, AIX, Linux, and Windows, el valor *receive time-out* se establece de la forma siguiente:

1. Para un número inicial de flujos, antes de que tenga lugar cualquier negociación, el valor de *tiempo de espera de recepción* es el doble del valor **HBINT** de la definición de canal.
2. Después de que los canales negocien un valor de **HBINT** , si **HBINT** se establece en menos de 60 segundos, el valor de *tiempo de espera de recepción* se establece en el doble de este valor. Si **HBINT** se establece en 60 segundos o más, el valor de *tiempo de espera de recepción* se establece en 60 segundos mayor que el valor de HBINT.

z/OS En z/OS, el valor de *tiempo de espera de recepción* se establece de la siguiente manera:

1. Para un número inicial de flujos, antes de que tenga lugar cualquier negociación, el valor de *tiempo de espera de recepción* es el doble del valor **HBINT** de la definición de canal.
2. Si se establece **RCVTIME** , el tiempo de espera se establece en uno de los valores siguientes, en función del parámetro **RCVTTYPE** , y sujeto a cualquier límite impuesto por **RCVTMIN** si se aplica:
   - del HBINT negociado multiplicado por una constante
   - del HBINT negociado más el número constante de segundos
   - de un número constante de segundos

   RCVTMIN no se aplica cuando **RCVTTYPE(EQUAL)** está configurado. Si utiliza un valor constante de **RCVTIME** y utiliza un intervalo de pulsaciones, no especifique un **RCVTIME** menor que el intervalo de pulsaciones. Para obtener detalles de los atributos **RCVTIME**, **RCVTMIN** y **RCVTTYPE** , consulte el mandato **ALTER QMGR** .

Nota:

1. Si cualquiera de los valores es cero, no hay tiempo de espera.
2. Para las conexiones que no dan soporte a pulsaciones, el valor **HBINT** se negocia en cero en el paso 2 y, por lo tanto, no hay tiempo de espera, por lo que debe utilizar `TCP/IP KEEPALIVE`.

3. Para conexiones de cliente que utilizan compartición de conversaciones, las pulsaciones pueden fluir por el canal (ambos extremos) todo el tiempo, no sólo cuando MQGET está pendiente.
4. Para conexiones de cliente en las que no se utiliza compartición de conversaciones, las pulsaciones fluyen del servidor únicamente cuando el cliente emite una llamada MQGET con espera. Por consiguiente, no se recomienda establecer el intervalo de pulsaciones demasiado bajo para canales de cliente. Por ejemplo, si el latido se establece en 10 segundos, una llamada MQCMIT falla (con MQRC_CONNECTION_BROKEN) si tarda más de 20 segundos en confirmarse porque no ha fluido ningún dato durante este tiempo. Esto puede suceder con grandes unidades de trabajo. Sin embargo, esto no ocurre si se eligen los valores apropiados para el intervalo de pulsaciones porque sólo MQGET con espera tarda períodos de tiempo significativos.

   Siempre que **SHARECNV** no sea cero, el cliente utiliza una conexión dúplex completa, lo que significa que el cliente puede (y lo hace) latido durante todas las llamadas MQI
5. Cancelar la conexión después del doble del intervalo de pulsaciones es válido porque se espera un flujo de datos o de pulsaciones como mínimo en cada intervalo de pulsaciones. No obstante, establecer un intervalo de pulsaciones demasiado bajo puede causar problemas, sobre todo si utiliza salidas de canal. Por ejemplo, si el valor de **HBINT** es un segundo y se utiliza una salida de envío o recepción, el extremo receptor espera sólo 2 segundos antes de cancelar el canal. Si el MCA está realizando una tarea como, por ejemplo, cifrar el mensaje, este valor puede ser demasiado corto.

Valores sugeridos

z/OS IBM MQ for z/OS

Como punto de partida inicial, puede utilizar:

```
/cpf ALTER QMGR TCPKEEP(YES) RCVTTYPE(ADD) RCVTIME(60) ADOPTMCA(ALL) ADOPTCHK(ALL)
```

donde `cpf` es el prefijo de mandato para el subsistema del gestor de colas.

Consulte **ALTER QMGR** y Disponibilidad de red de IBM MQ para obtener más información sobre los diversos parámetros.

Si la dirección IP del remitente puede convertirse en más de una dirección, es posible que tenga que establecer **ADOPTCHK** en QMNAME en lugar de ALL.

Multi IBM MQ for Multiplatforms

En `qm.ini`, añada la información siguiente:

```
TCP:
KeepAlive=Yes
CHANNELS:
AdoptNewMCA=ALL
AdoptNewMCACheck=ALL
```

Consulte **ALTER QMGR**, stanzas del archivo de configuración para gestión de colas distribuidasy “Stanza de canales del archivo qm.ini” en la página 129 para obtener más información.

Si la dirección IP del remitente puede convertirse en más de una dirección, es posible que tenga que establecer **AdoptNewMCACheck** en QMNAME en lugar de **ALL**.

Adoptar un MCA

La función de adopción de un MCA permite a IBM MQ cancelar un canal receptor e iniciar uno nuevo en su lugar.

Si un canal pierde el contacto, el canal receptor puede quedar en un estado de 'recepción de comunicaciones'. Cuando se restablecen las comunicaciones el canal emisor intenta reconectarse. Si el gestor de colas remoto descubre que el canal receptor ya está en ejecución, no permite que se inicie

otra versión del mismo canal receptor. Este problema requiere la intervención del usuario para rectificar el problema o el uso de mantenimiento del sistema.

La función de Adoptar MCA soluciona el problema automáticamente. Permite a IBM MQ cancelar un canal receptor e iniciar uno nuevo en su lugar.

Tareas relacionadas

Administración de IBM MQ

z/OS Administración de IBM MQ for z/OS

IBM i Administración de IBM MQ for IBM i

Detención y desactivación temporal de canales

Puede detener y desactivar temporalmente un canal antes de que caduque el intervalo de tiempo de desconexión.

Los canales de mensajes están diseñados para ser conexiones duraderas entre gestores de colas con una terminación ordenada que únicamente controla el atributo de canal de intervalo de desconexión. Este mecanismo funciona bien a menos que el operador debe terminar el canal antes de que caduque el intervalo de tiempo de desconexión. Esto debe producirse en las situaciones siguientes:

- Inmovilización del sistema
- Conservación de recursos
- Acción unilateral en un extremo del canal

En este caso, puede detener el canal. Puede hacerlo utilizando:

- El mandato STOP CHANNEL MQSC
- El mandato Detener canal PCF
- IBM MQ Explorer
- z/OS IBM i otros mecanismos específicos de la plataforma, tal como se indica a continuación:

  z/OS **Para z/OS:**
  El panel Detener un canal

  IBM i **Para IBM i:**
  El mandato ENDMQMCHL CL o la opción END en el panel WRKMQMCHL

Hay tres opciones para detener los canales utilizando estos mandatos:

QUIESCE
La opción QUIESCE intenta finalizar el lote actual de mensajes antes de detener el canal.

FORCE
La opción FORCE intenta detener el canal inmediatamente y puede precisar que el canal se resincronice cuando se reinicie porque el canal puede quedar pendiente.

z/OS En IBM MQ for z/OS, FORCE interrumpe cualquier reasignación de mensajes en curso, lo cual puede dejar mensajes BIND_NOT_FIXED reasignados parcialmente o dañados.

TERMINATE
La opción TERMINATE intenta detener el canal inmediatamente y termina la hebra o el proceso del canal.

z/OS En IBM MQ for z/OS, TERMINATE interrumpe cualquier reasignación de mensajes en curso, lo cual puede dejar mensajes BIND_NOT_FIXED reasignados parcialmente o dañados.

Todas estas opciones dejar el canal en un estado STOPPED que requiere la intervención del operador para reiniciarlo.

Detener el canal en el extremo emisor es efectivo pero no requiere la intervención del operador para reiniciarse. En el extremo receptor del canal, las cosas son mucho más difíciles debido a que el MCA está a la espera de datos del área de emisión y no hay modo de iniciar una terminación *ordenada* del canal desde el área de recepción; el mandato stop está pendiente hasta que el MCA retorno de la espera de datos.

Por consiguiente, hay tres formas recomendadas de utilizar canales, en función de las características operativas necesarias:

- Si desea que los canales sean de ejecución prolongada, tenga en cuenta que sólo puede haber una terminación ordenada desde el extremo emisor. Cuando los canales se interrumpen, es decir, se detienen, se requiere la intervención del operador (un mandato START CHANNEL) con objeto de reiniciarlos.
- Si desea que los canales estén activos sólo cuando haya mensajes para transmitir, establezca el intervalo de desconexión en un valor realmente bajo. El valor predeterminado es alto y, por consiguiente, no se recomienda para canales donde se requiere este nivel de control. Puesto que resulta difícil interrumpir el canal receptor, la opción más económica es hacer que el canal se desconecte y se vuelva a conectar automáticamente según lo exija la carga de trabajo. Para la mayoría de los canales, el valor adecuado del intervalo de desconexión se puede establecer de forma heurística.
- Puede utilizar el atributo de intervalo de pulsaciones para hacer que el MCA envíe un flujo de pulsaciones al MCA receptor durante periodos en los que no hay mensajes que enviar. Esta acción libera el MCA receptor de su estado de espera y le brinda la oportunidad de desactivar temporalmente el canal sin esperar a que el intervalo de desconexión caduque. Dé al intervalo de pulsaciones un valor más bajo que el del intervalo de desconexión.

  Nota:

  1. Es aconsejable para establecer el intervalo de desconexión en un valor bajo o utilizar pulsaciones, para canales de servidor. Este valor bajo es para permitir en el caso de que el canal peticionario finalice de forma anómala (por ejemplo porque se canceló el canal) cuando no hay mensajes del canal servidor que enviar. Si el intervalo de desconexión se establece alto y no se utilizan pulsaciones, el servidor no detecta que el peticionario ha finalizado (cosa que sólo hará la próxima vez que intente enviar un mensaje al peticionario). Mientras el servidor sigue en ejecución, tiene la cola de transmisión abierta para entrada exclusiva con objeto de obtener cualquier mensaje adicional llegue a la cola. Si se intenta reiniciar el canal del peticionario, la solicitud de inicio recibe un error porque el servidor sigue teniendo la cola de transmisión abierta para entrada exclusiva. Es necesario detener el canal servidor y, a continuación, reiniciar el canal desde el peticionario de nuevo.

Reinicio de canales detenidos

Cuando un canal pasa al estado STOPPED, es preciso que reinicie el canal manualmente.

Acerca de esta tarea

Para canales emisores o servidores, cuando el canal ha entrado en el estado STOPPED, la cola de transmisión asociada se ha establecido en GET(DISABLED) y se ha desactivado el desencadenamiento. Cuando se recibe la solicitud de inicio, estos atributos se restablecen automáticamente.

z/OS Si el iniciador de canal se detiene mientras un canal está en estado RETRYING o STOPPED, el estado de canal se recuerda cuando se reinicia el iniciador de canal. Sin embargo, el estado del canal para el tipo de canal SVRCONN se restablece si el iniciador de canal se detiene mientras el canal está en estado STOPPED.

Multi Si el gestor de colas se detiene mientras un canal está en estado RETRYING o STOPPED, el estado de canal se recuerda cuando se reinicia el gestor de colas. A partir de IBM MQ 8.0, estos se aplica también a los canales SVRCONN. Anteriormente, el estado de canal para el tipo de canal SVRCONN se ha restablecido si el iniciador de canal se ha detenido mientras el canal estaba en estado STOPPED.

Procedimiento

- Reinicie el canal de una de las siguientes maneras:
  - Con el Comando START CHANNEL MQSC.
  - Con el Comando PCF Start Channel.
  - Con IBM MQ Explorer
  - z/OS En z/OS, con el Panel Iniciar un canal.
  - IBM i En IBM i, con el Comando CL STRMQMCHL o con la opción START del Panel WRKMQMCHL.

Manejo de canales pendientes

Un canal pendiente es un canal que está pendiente con un canal remoto de los mensajes que se han enviado y recibido.

Acerca de esta tarea

Observe la diferencia entre esto y un gestor de colas que está pendiente de qué mensajes se deben confirmar en una cola.

Puede reducir la posibilidad de que un canal se ponga en duda utilizando el parámetro de canal de latido por lotes (**BATCHHB**). Cuando se especifica un valor para este parámetro, un canal emisor comprueba que el canal remoto sigue activo antes de realizar cualquier otra acción. Si no se recibe ninguna respuesta del canal receptor, se considera que ya no está activo. Los mensajes se pueden restituir y redirigir, y el canal emisor deja de estar pendiente. Esto reduce el tiempo que el canal puede estar pendiente al periodo que transcurre entre el canal emisor que verifica si el canal receptor sigue estando activo y verificar si el canal receptor ha recibido los mensajes enviados. Consulte la sección Atributos de canal para obtener más información sobre el parámetro de pulsaciones por lotes.

Los problemas de canal pendientes suelen resolverse automáticamente. Incluso cuando se pierde la comunicación y un canal se coloca en estado pendiente con un lote de mensajes en el emisor con estado de recepción desconocido, la situación se resuelve cuando se restablece la comunicación. El número de secuencia y los registros LUWID se mantienen para este fin. El canal está pendiente hasta que se ha intercambiado información LUWID y sólo puede haber un lote de mensajes pendiente para el canal.

Puede, cuando sea necesario, resincronizar el canal manualmente. El término manual incluye el uso de operadores o programas que contienen mandatos de gestión del sistema IBM MQ . El proceso de resincronización manual funciona de la manera siguiente. Esta descripción utiliza mandatos MQSC, pero también puede utilizar los equivalentes PCF.

Procedimiento

1. Utilice el mandato **DISPLAY CHSTATUS** para buscar el último ID de unidad lógica de trabajo (LUWID) confirmado para cada lado del canal.

   Para ello, utilice los mandatos siguientes:

   - Para el área pendiente del canal:

     ```
     DISPLAY CHSTATUS(name) SAVED CURLUWID
     ```

     Puede utilizar los parámetros **CONNAME** y **XMITQ** para identificar más el canal.
   - Para el área de recepción del canal:

     ```
     DISPLAY CHSTATUS( name ) SAVED LSTLUWID
     ```

     Puede utilizar el parámetro **CONNAME** para identificar más el canal.

Nota: Los mandatos son diferentes debido a que sólo el área de emisión del canal puede estar pendiente. El área de emisión nunca está pendiente.

IBM i En IBM i, el mandato **DISPLAY CHSTATUS** se puede ejecutar desde un archivo utilizando el mandato **STRMQMMQSC** o el mandato CL Trabajar con estado de canal MQM, **WRKMQMCHST**.

2. Si los dos LUWID son iguales, utilice el mandato **RESOLVE CHANNEL** para confirmar los mensajes pendientes.

   Si los dos LUWID son iguales, el área de recepción ha confirmado la unidad de trabajo que el emisor considera pendiente. El área de emisión puede ahora eliminar los mensajes pendientes de la cola de transmisión y rehabilitarlos. Esto se realiza con el siguiente mandato **RESOLVE CHANNEL** :

   ```
   RESOLVE CHANNEL(name) ACTION(COMMIT)
   ```

3. Si los dos LUWID son diferentes, utilice el mandato **RESOLVE CHANNEL** para restituir los mensajes pendientes.

   Si los dos luwid son diferentes, el área de recepción no ha confirmado la unidad de trabajo que el emisor considera pendiente. El área de emisión debe retener los mensajes pendientes en la cola de transmisión y reenviarlos. Esto se realiza con el siguiente mandato **RESOLVE CHANNEL** :

   ```
   RESOLVE CHANNEL( name ) ACTION(BACKOUT)
   ```

IBM i En IBM i, puede utilizar el mandato Resolver canal MQM, **RSVMQMCHL**.

Resultados

Una vez que este proceso se haya completado el canal ya no está pendiente. Otro canal puede utilizar la cola de transmisión si es preciso.

Referencia relacionada

DISPLAY CHSTATUS (visualizar estado de canal)

RESOLVE CHANNEL (solicitar a un canal que resuelva mensajes pendientes)

Seguridad de mensajes

Además de las funciones de recuperación habituales de IBM MQ, la gestión de colas distribuidas garantiza la entrega correcta de los mensajes utilizando un procedimiento de punto de sincronización coordinado entre los dos extremos del canal de mensajes. Si este procedimiento detecta un error, cierra el canal para que pueda investigar el problema y mantiene los mensajes de forma segura en la cola de transmisión hasta que se reinicia el canal.

El procedimiento de punto de sincronización tiene una ventaja añadida, ya que intenta recuperar una situación *pendiente* cuando se inicia el canal. (*Pendiente* es el estado de una unidad de recuperación para la que se ha solicitado un punto de sincronización pero todavía no se conoce el resultado de la solicitud.) Con este recurso también están asociadas estas dos funciones:

1. Resolver con confirmación o restitución
2. Restablecer el número de secuencia

El uso de estas funciones se produce sólo en circunstancias excepcionales porque el canal se recupera automáticamente en la mayoría de los casos.

Mensajes rápidos no persistentes

El atributo de canal de velocidad de mensajes no persistentes (NPMSPEED) se puede utilizar para especificar que los mensajes no persistentes en el canal deben entregarse más rápidamente. Para obtener más información sobre este atributo, consulte Velocidad de mensajes no persistentes (NPMSPEED).

Si un canal termina mientras existen mensajes rápidos no persistentes en tránsito, éstos se pueden perder y corresponde a la aplicación recuperarlos si es necesario.

Si el canal receptor no puede colocar el mensaje en la cola de destino, se pone en la cola de mensajes no entregados, si se ha definido una. De lo contrario, el mensaje se descarta.

Nota: Si el otro extremo del canal no admite la opción, el canal se ejecuta a velocidad normal.

Mensajes no entregados

Para obtener información sobre lo que ocurre cuando no puede entregarse un mensaje, consulte "¿Qué sucede cuando no puede entregarse un mensaje?" en la página 254.

¿Qué sucede cuando no puede entregarse un mensaje?

Cuando un mensaje no puede entregarse, el MCA puede procesarlo de varias formas. Puede intentarlo de nuevo, puede devolvérselo al emisor o puede ponerlo en la cola de mensajes no entregados.

En la Figura 23 en la página 254 se muestra el proceso que tiene lugar cuando un MCA no puede poner un mensaje en la cola de destino. (Las opciones mostradas no se aplican a todas las plataformas).

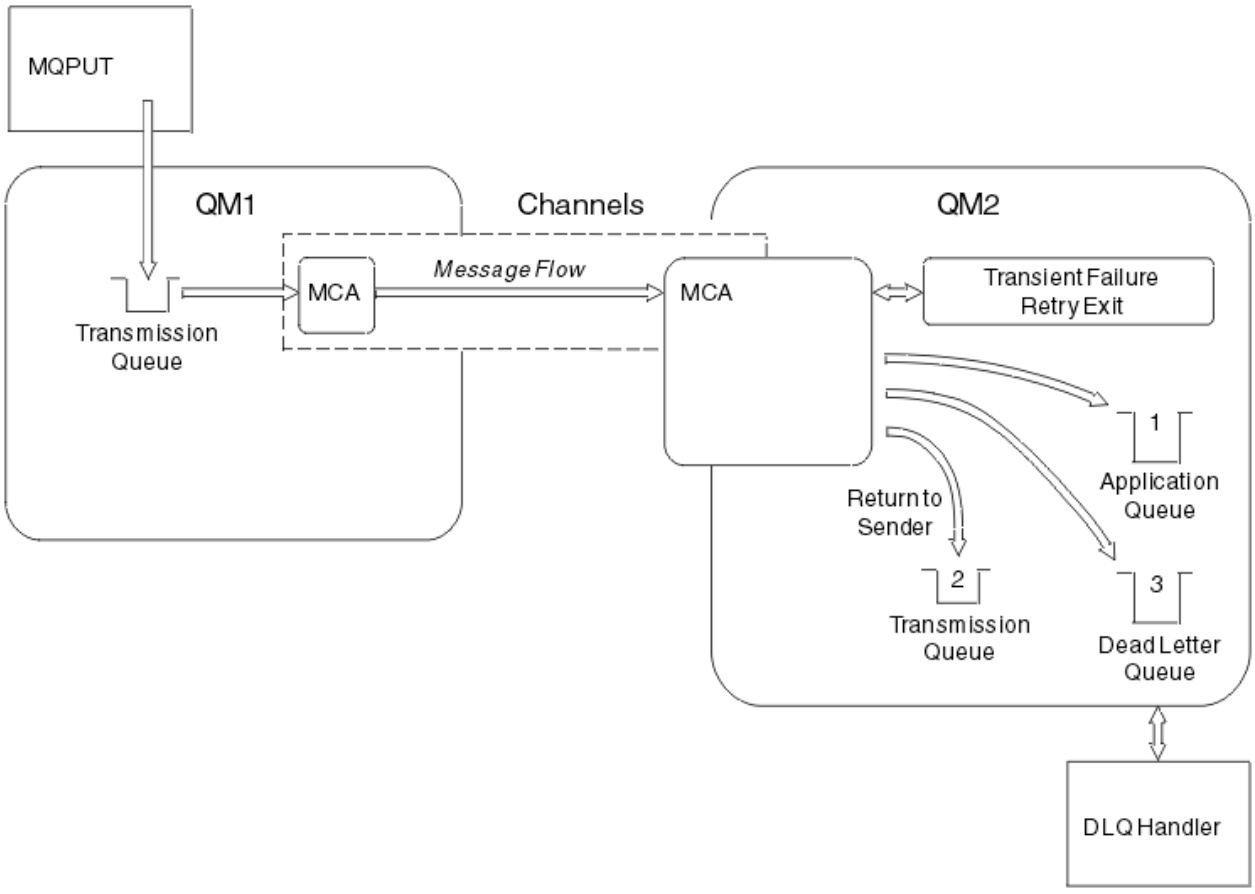


Figura 23. Qué ocurre cuando un mensaje no puede entregarse

Tal como se muestra en la figura, el MCA puede hacer varias cosas con un mensaje que no puede entregar. La acción está determinada por las opciones especificadas cuando se define el canal y por las opciones de informe MQPUT del mensaje.

1. Reintento de mensaje

   Si el MCA no puede poner un mensaje en la cola de destino por una razón que puede ser pasajera (por ejemplo, que la cola está llena), el MCA puede esperar y volver a intentar la operación más adelante. Puede determinar si el MCA esperará, durante cuánto tiempo y cuántas veces volverá a intentar la operación.

- Al definir el canal puede especificar un tiempo y un intervalo de reintento de mensaje para los errores MQPUT. Si el mensaje no puede transferirse a la cola de destino porque la cola está llena o inhibida para transferencias, el MCA intenta la operación el número de veces especificado, en el intervalo de tiempo especificado.
- Puede escribir su propia salida de reintento de mensaje. La salida permite especificar las condiciones en que el MCA volverá a intentar la operación MQPUT o MQOPEN. Especifique el nombre de la salida al definir el canal.

2. Devolución al emisor

   Si el reintento de mensaje no ha tenido éxito o se ha producido otro tipo de error, el MCA puede devolver el mensaje al originador. Para habilitar la capacidad de devolver al emisor, debe especificar las siguientes opciones en el descriptor de mensaje al poner el mensaje en la cola original:

   - La opción de informe MQRO_EXCEPTION_WITH_FULL_DATA
   - La opción de informe MQRO_DISCARD_MSG
   - El nombre de la cola de respuesta y el gestor de colas de respuesta

   Si el MCA es capaz de colocar el mensaje en la cola de destino, genera un informe de excepción que contiene el mensaje original y lo pone en una cola de transmisión para enviarlo a la cola de respuesta especificado en el mensaje original. (Si la cola de respuesta se encuentra en el mismo gestor de colas que el MCA, el mensaje se transfiere directamente a esa cola, no a una cola de transmisión).

3. Cola de mensajes no entregados

   Si no se puede entregar o devolver un mensaje, se coloca en la cola de mensajes no entregados (DLQ). Puede utilizar el manejador DLQ para procesar el mensaje. Este proceso se describe aquí:

   - ULW Proceso de mensajes en una cola de mensajes no entregados
   - z/OS Programa de utilidad de manejador de colas de mensajes no entregados (CSQUDLQH)

   Si la cola de mensajes no entregados no está disponible, el MCA emisor deja el mensaje en la cola de transmisión y el canal se detiene. En un canal rápido, los mensajes no persistentes que no pueden escribirse en una cola de mensajes no entregados se pierden.

   En IBM WebSphere MQ 7.0, si no se define una cola de mensajes no entregados local, la cola remota no está disponible o no se ha definido y no hay ninguna cola de mensajes no entregados remota, entonces el canal emisor se coloca en estado de REINTENTO y los mensajes se restituyen automáticamente a la cola de transmisión.

Referencia relacionada

Utilización de la cola de mensajes no entregados (USEDLQ)

Desencadenamiento de canales

IBM MQ proporciona un recurso para iniciar una aplicación automáticamente cuando se cumplen ciertas condiciones en una cola. Este recurso se denomina desencadenamiento.

Esta explicación está pensada como una visión general de los conceptos de desencadenamiento. Para obtener una descripción completa, consulte Inicio de aplicaciones IBM MQ utilizando desencadenantes.

Para información específica de la plataforma, consulte:

- Para AIX, Linux, and Windows, consulte “Desencadenamiento de canales en AIX, Linux, and Windows.” en la página 257
- IBM i Para IBM i, consulte “Desencadenamiento de canales en IBM MQ for IBM i” en la página 257
- z/OS Para z/OS, consulte “Transmission queues and triggering channels” en la página 978

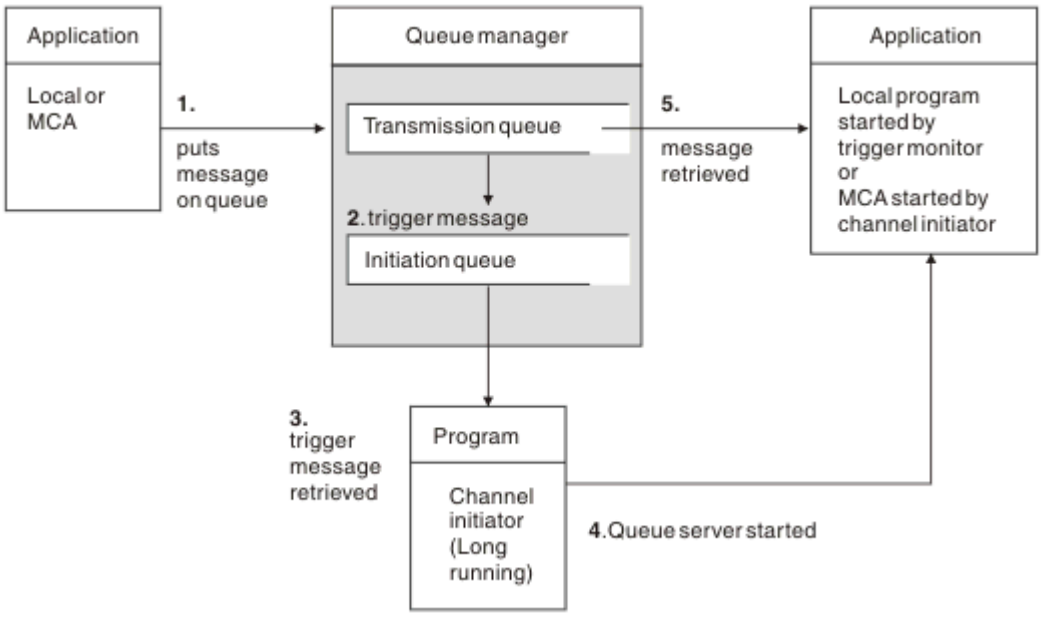


Figura 24. Conceptos de desencadenamiento

Los objetos necesarios para el desencadenamiento se muestran en la Figura 24 en la página 256. Muestra la siguiente secuencia de sucesos:

1. El gestor de colas local coloca un mensaje de una aplicación o de un agente de canal de mensajes (MCA) en la cola de transmisión.
2. Cuando las condiciones de desencadenamiento se cumplen, el gestor de colas local coloca un mensaje desencadenante en la cola de inicio.
3. El programa iniciador de canal supervisa la cola de inicio y recupera los mensajes a medida que llegan.
4. El iniciador de canal procesa los mensajes desencadenantes de acuerdo con la información contenida en ellos. Esta información puede incluir el nombre de canal, en cuyo caso se inicia el MCA correspondiente.
5. La aplicación local o el MCA, habiendo sido desencadenado, recupera los mensajes de la cola de transmisión.

Para configurar este caso práctico, tendrá que:

- Crear la cola de transmisión con el nombre de la cola de inicio (es decir, SYSTEM.CHANNEL.INITQ) en el atributo correspondiente.
- Asegurarse de que la cola de inicio (SYSTEM.CHANNEL.INITQ) existe.
- Asegurarse de que el programa iniciador de canal está disponible y en ejecución. El programa iniciador de canal ha de ejecutarse siempre con el nombre de la cola de inicio en su mandato de inicio.
  z/OS En z/OS, el nombre de la cola de inicio es fijo, por lo que no se utiliza en el mandato de inicio.
- Opcionalmente, cree la definición de proceso para el mecanismo de desencadenamiento, si no existe, y asegúrese de que el campo *UserData* contiene el nombre del canal al que sirve. En vez de crear una definición de proceso, puede especificar el nombre de canal en el atributo **TriggerData** de la cola de transmisión. IBM MQ para sistemas IBM i IBM i, AIX, Linux, and Windows, permite que el nombre del canal se especifique en blanco, en cuyo caso se utiliza la primera definición de canal disponible con esta cola de transmisión.
- Asegúrese de que la definición de la cola de transmisión contiene el nombre de la definición de proceso a la que servir (si es aplicable), el nombre de la cola de inicio y las características de

desencadenamiento que considere más adecuadas. El atributo de control desencadenante permite habilitar o inhabilitar el mecanismo de desencadenamiento, según convenga.

Nota:

1. El programa iniciador de canal actúa como un 'supervisor desencadenante' supervisando la cola de iniciación utilizada para iniciar canales.
2. Puede utilizarse una cola de inicio y el proceso desencadenante para desencadenar cualquier número de canales.
3. Puede definirse cualquier número de colas de inicio y de procesos desencadenantes.
4. Se recomienda el tipo de desencadenante FIRST para evitar inundar el sistema con inicios de canal.

Desencadenamiento de canales en AIX, Linux, and Windows.

ALW

Puede crear una definición de proceso en IBM MQ, definiendo procesos para desencadenar. Utilice el mandato MQSC DEFINE PROCESS para crear una definición de proceso que denomine el proceso que se desencadenará cuando lleguen mensajes a una cola de transmisión. El atributo USERDATA de la definición de proceso contiene el nombre del canal al que da servicio la cola de transmisión.

Defina la cola local (QM4), especificando que los mensajes desencadenantes han de escribirse en la cola de inicio (IQ) para desencadenar la aplicación que inicia el canal (QM3.TO.QM4):

```
DEFINE QLOCAL(QM4) TRIGGER INITQ(SYSTEM.CHANNEL.INITQ) PROCESS(P1) USAGE(XMITQ)
```

Defina la aplicación (proceso P1) que ha de iniciarse:

```
DEFINE PROCESS(P1) USERDATA(QM3.TO.QM4)
```

De forma alternativa, para IBM MQ for UNIX y sistemas Linux y Windows, puede eliminar la necesidad de una definición de proceso especificando el nombre de canal en el atributo TRIGDATA de la cola de transmisión.

Defina la cola local (QM4). Especifique que los mensajes desencadenantes se escriben en la cola de inicio predeterminada SYSTEM.CHANNEL.INITQ para desencadenar la aplicación (proceso P1) que inicia el canal (QM3.TO.QM4):

```
DEFINE QLOCAL(QM4) TRIGGER INITQ(SYSTEM.CHANNEL.INITQ)
USAGE(XMITQ) TRIGDATA(QM3.TO.QM4)
```

Si no especifica un nombre de canal, el iniciador de canal busca los archivos de definición de canal hasta que encuentre un canal que esté asociado a la cola de transmisión designada.

Desencadenamiento de canales en IBM MQ for IBM i

IBM i

El desencadenamiento de canales en IBM MQ for IBM i se implementa con el proceso iniciador de canal. Un proceso iniciador de canal para la cola de inicio SYSTEM.CHANNEL.INITQ se inicia automáticamente con el gestor de colas a menos que se inhabilite modificando el atributo SCHINIT del gestor de colas.

Configure la cola de transmisión para el canal, especificando SYSTEM.CHANNEL.INITQ como cola de inicio y habilitando el mecanismo de desencadenamiento para la cola. El iniciador de canal inicia el primer canal disponible que especifique esta cola de transmisión.

```
CRTMQMQ QNAME(MYXMITQ1) QTYPE(*LCL) MQMNAME(MYQMGR)
```

```
TRGENBL(*YES) INITQNAME(SYSTEM.CHANNEL.INITQ)
USAGE(*TMQ)
```

Deprecated Puede iniciar manualmente hasta tres procesos iniciadores de canal con el mandato STRMQMCHLI y especificar colas de inicio distintas. También puede especificar más de un canal capaz de procesar la cola de transmisión y elegir qué canal iniciar. Esta posibilidad se sigue ofreciendo por motivos de compatibilidad con releases anteriores. Está en desuso.

Nota: Sólo un canal a la vez puede procesar una cola de transmisión.

```
STRMQMCHLI QNAME(MYINITQ)
```

Configure la cola de transmisión para el canal, especificando TRGENBL(*YES) y, para elegir qué canal se intentará iniciar, especifique el nombre del canal en el campo TRIGDATA. Por ejemplo:

```
CRTMQMQ QNAME(MYXMITQ2) QTYPE(*LCL) MQMNAME(MYQMGR)
TRGENBL(*YES) INITQNAME(MYINITQ)
USAGE(*TMQ) TRIGDATA(MYCHANNEL)
```

Conceptos relacionados

“Inicio y detención del iniciador de canal” en la página 258
El desencadenamiento se implementa utilizando el proceso de iniciador de canal.

Tareas relacionadas

“Configuración de la gestión de colas distribuidas” en la página 210
En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

Referencia relacionada

Programas de canal en AIX, Linux, and Windows

IBM i Trabajos de intercomunicación en IBM i

IBM i Estados de canal en IBM i

Inicio y detención del iniciador de canal

El desencadenamiento se implementa utilizando el proceso de iniciador de canal.

Este proceso de iniciador de canal se inicia con el mandato de MQSC START CHINIT. A menos que esté utilizando la cola de inicio predeterminada, especifique el nombre de la cola de inicio en el mandato. Por ejemplo, para utilizar el mandato START CHINIT para iniciar la cola IQ para el gestor de colas predeterminado, entre:

```
START CHINIT INITQ(IQ)
```

De forma predeterminada, un iniciador de canal se inicia automáticamente utilizando la cola de inicio predeterminada, SYSTEM.CHANNEL.INITQ. Si desea iniciar todos los iniciadores de canal manualmente, siga estos pasos:

1. Cree e inicie el gestor de colas.
2. Altere la propiedad SCHINIT del gestor de colas en MANUAL
3. Finalice y reinicie el gestor de colas

En sistemas IBM MQ for Multiplatforms, se inicia automáticamente un iniciador de canal. El número de iniciadores de canal que puede iniciar es limitado. El valor predeterminado y máximo es 3. Puede cambiarlo utilizando MAXINITIATORS en el archivo qm.ini para sistemas AIX and Linux y en el registro para sistemas Windows.

Consulte Mandatos de control de IBM MQ para obtener información detallada sobre el mandato ejecutar iniciador de canal **runmqchi** y los demás mandatos de control.

Detención del iniciador de canal

El iniciador de canal predeterminado se inicia automáticamente cuando se inicia un gestor de colas. Todos los iniciadores de canal se detienen automáticamente cuando se detiene un gestor de colas.

Archivos de inicialización y configuración

El manejo de los datos de inicialización del canal depende de la plataforma de IBM MQ.

IBM MQ for z/OS

En IBM MQ for z/OS, la información de inicialización y configuración se especifica utilizando el mandato MQSC **ALTER QMGR** . Si coloca mandatos **ALTER QMGR** en el conjunto de datos de entrada de inicialización CSQINP2 , se procesan cada vez que se inicia el gestor de colas.

Para ejecutar mandatos MQSC como **START LISTENER** cada vez que inicie el iniciador de canal, pónselos en el conjunto de datos de entrada de inicialización CSQINPX y especifique la sentencia DD opcional CSQINPX en el procedimiento de tarea iniciada del iniciador de canal.

Para obtener más información sobre CSQINP2 y CSQINPX, consulte Personalizar los conjuntos de datos de entrada de inicializacióny ALTER QMGR.

IBM MQ for Multiplatforms

En IBM MQ for Multiplatforms, hay archivos de configuración que contienen información de configuración básica sobre la instalación de IBM MQ .

Hay dos archivos de configuración: uno se aplica a la máquina, el otro se aplica a un gestor de colas individual.

Archivo de configuración de IBM MQ
Este archivo contiene información relacionada con todos los gestores de colas en el sistema IBM MQ. El archivo se llama `mqs.ini`. Está descrito en “Archivo de configuración de IBM MQ, mqs.ini” en la página 96.

Archivo de configuración del gestor de colas

Este archivo contiene información de configuración relacionada con un determinado gestor de colas. El archivo se llama `qm.ini`.

Se crea durante la creación del gestor de colas y puede contener información de configuración relacionada con cualquier aspecto del gestor de colas. La información contenida en el archivo incluye detalles de cómo la configuración del registro difiere de la predeterminada en el archivo de configuración de IBM MQ.

El archivo de configuración del gestor de colas se encuentra en la raíz del árbol de directorios que ocupa el gestor de colas. Por ejemplo, para los atributos de **DefaultPath**, los archivos de configuración del gestor de colas para un gestor de colas llamado QMNAME serían:

Para sistemas AIX and Linux:

```
/var/mqm/qmgrs/QMNAME/qm.ini
```

Para sistemas Windows:

```
C:\ProgramData \IBM \MQ\qmgrs\QMNAME\qm.ini
```

IBM i Para IBM i:

```
/QIBM/UserData/mqm/qmgrs/QMNAME/qm.ini
```

Aquí hay un extracto de un `qm.ini`. Especifica que el escucha TCP/IP debe escuchar en el puerto 2500, el número máximo de canales actuales es 200 y el número máximo de canales activos es 100.

```
TCP:
Port=2500
CHANNELS:
MaxChannels=200
MaxActiveChannels=100
```

Puede especificar un rango de puertos TCP/IP que utilizará un canal de salida. Un método es utilizar el archivo `qm.ini` para especificar el inicio y el final de un rango de valores de puerto. El ejemplo siguiente muestra un archivo `qm.ini` que especifica un rango de canales:

```
TCP:
StrPort=2500
EndPort=3000
CHANNELS:
MaxChannels=200
MaxActiveChannels=100
```

Si especifica un valor para **StrPort** o **EndPort**, debe especificar un valor para ambos. El valor de **EndPort** siempre debe ser mayor que el valor de **StrPort**.

El canal intenta utilizar cada uno de los valores de puerto en el rango especificado. Cuando la conexión se realiza correctamente, el valor del puerto es el puerto que el canal utiliza.

Para obtener más información sobre los archivos `qm.ini` , consulte “Archivos de configuración de gestores de colas, qm.ini” en la página 109.

Conversión de datos para mensajes

Los mensajes de IBM MQ podrían necesitar la conversión de datos cuando se envían entre colas en distintos gestores de colas.

Un mensaje de IBM MQ consta de dos partes:

- Información de control en un descriptor de mensaje
- Datos de la aplicación

Cualquiera de las dos partes puede requerir la conversión de datos cuando se envían entre colas en gestores de colas diferentes. Si desea más información sobre la conversión de datos de aplicación, consulte Conversión de datos de aplicación.

Escribir sus propios agentes de canales de mensajes

IBM MQ le permite escribir sus propios programas de agente de canal de mensajes (MCA) o instalar el de un proveedor de software independiente.

Puede que le interese escribir sus propios programas MCA para que IBM MQ interactúe con un protocolo de comunicaciones de propiedad, o para enviar mensajes mediante un protocolo al que IBM MQ no da soporte. (No puede escribir su propio MCA para interactuar con un MCA suministrado por IBM MQ en el otro extremo.)

Si decide utilizar un MCA que no ha sido suministrado por IBM MQ deberá tener en cuenta las siguientes cuestiones.

Envío y recepción de mensajes
Debe escribir una aplicación emisora que obtenga los mensajes de allí donde los coloque la aplicación, por ejemplo de una cola de transmisión, y los envíe en el protocolo con el que desee

comunicarse. También debe escribir una aplicación receptora que tome los mensajes de este protocolo y los coloque en las colas de destino. Las aplicaciones de envío y recepción utilizan las llamadas MQI (interfaz de colas de mensajes), no las de interfaces especiales.

Debe asegurarse de que los mensajes sólo se entregan una vez. para ayudar en esta entrega se puede utilizar la coordinación del punto de sincronización.

Función de control de canales
Debe proporcionar sus propias funciones de administración para controlar los canales. No puede utilizar las funciones de administración de IBM MQ para configurar (por ejemplo, el mandato DEFINE CHANNEL) o supervisar (por ejemplo, DISPLAY CHSTATUS) los canales.

Archivo de inicialización
Debe proporcionar su propio archivo de inicialización, si necesita uno.

Conversión de datos de aplicación
Probablemente le interese permitir la conversión de datos de los mensajes que envíe a un sistema diferente. En tal caso, utilice la opción MQGMO_CONVERT en la llamada MQGET cuando recupere mensajes de allí donde los coloque la aplicación, por ejemplo de la cola de transmisión.

Salidas de usuario
Considere si necesita salidas de usuario. Si es así, puede utilizar las mismas definiciones de interfaz que las que utiliza IBM MQ.

Desencadenamiento
Si la aplicación transfiere los mensajes a una cola de transmisión, puede configurar los atributos de la cola de transmisión de tal modo que el MCA emisor se active cuando los mensajes lleguen a la cola.

Iniciador de canal
Tal vez deba proporcionar su propio iniciador de canal.

Otras cosas que hay que tener en cuenta para gestionar colas distribuidas

Otros temas que hay que tener en cuenta cuando se prepara IBM MQ para la gestión de colas distribuidas. Este tema cubre las colas de mensajes no entregados, las colas en uso, las extensiones del sistema y los programas de salida de usuario, y la ejecución de canales y escuchas como aplicaciones de confianza.

Cola de mensajes no entregados

Para asegurarnos de que los mensajes que llegan a la cola de mensajes no entregados (también conocida como DLQ) se procesan, cree un programa que pueda desencadenarse o ejecutarse a intervalos regulares para manejar estos mensajes.

Linux AIX Se proporciona un manejador DLQ con IBM MQ en sistemas AIX and Linux; para obtener más información, consulte El manejador DLQ de ejemplo, amqsdlq.

IBM i Para obtener más información sobre IBM MQ for IBM i, consulte El manejador de la cola de mensajes no entregados de IBM MQ for IBM i.

Colas en uso

Los MCA para canales receptores pueden mantener abiertas las colas de destino incluso cuando no se transmiten mensajes. Esto tiene como consecuencia que las colas parece que están "en uso".

Número máximo de canales

IBM i En IBM MQ for IBM i puede especificar el número máximo de canales permitidos en el sistema y el número máximo que pueden estar activos al mismo tiempo. Especifique estos números en el archivo `qm.ini` del directorio QIBM/UserData/mqm/qmgrs/*nombre_gestor_colas*. Consulte Stanzas del archivo de configuración para la gestión de colas distribuidas.

Extensiones del sistema y programas de salida de usuario

En la definición de canal se proporciona un recurso para permitir que se ejecuten programas adicionales en momentos concretos durante el proceso de mensajes. Estos programas no se suministran con IBM MQ, pero pueden proporcionarse en cada instalación en función de los requisitos locales.

Para poder ejecutarse, estos programas de salida de usuario debe tener nombres predefinidos y estar siempre disponibles para los programas de canal. Los nombres de los programas de salida de usuario se incluyen en las definiciones de canal de mensajes.

Se define una interfaz de bloque de control para entregar el control a estos programas y para manejar la devolución del control de estos programas.

Los lugares precisos donde se llaman estos programas y detalles sobre los bloques de control y los nombres se encuentran en Programas de salida de canal para canales de mensajería.

Ejecución de canales y escuchas como aplicaciones de confianza

Si el rendimiento es importante en el entorno y éste es estable, puede ejecutar los canales y escuchas como de confianza, utilizando el enlace FASTPATH. Hay dos factores que influyen en si los canales y escuchas se ejecutan como de confianza:

- La variable de entorno MQ_CONNECT_TYPE=FASTPATH o MQ_CONNECT_TYPE=STANDARD. Hace distinción entre mayúsculas y minúsculas. Si especifica un valor que no es válido, se pasará por alto.
- MQIBindType en la stanza Channels de `qm.ini` o del archivo de registro. Puede establecerlo en FASTPATH o STANDARD y no hace distinción entre mayúsculas y minúsculas. El valor predeterminado es STANDARD.

Puede utilizar MQIBindType en asociación con la variable de entorno para obtener el efecto deseado, del modo siguiente:

MQIBindType	**Variable de entorno**	**Resultado**
ESTÁNDAR	UNDEFINED	ESTÁNDAR
FASTPATH	UNDEFINED	FASTPATH
ESTÁNDAR	ESTÁNDAR	ESTÁNDAR
FASTPATH	ESTÁNDAR	ESTÁNDAR
ESTÁNDAR	FASTPATH	ESTÁNDAR
FASTPATH	FASTPATH	FASTPATH
ESTÁNDAR	CLIENTE	CLIENTE
FASTPATH	CLIENTE	ESTÁNDAR
ESTÁNDAR	LOCAL	ESTÁNDAR
FASTPATH	LOCAL	ESTÁNDAR

En resumen, sólo hay dos formas de que los canales y escuchas se ejecuten realmente como de confianza:

1. Especificando MQIBindType=FASTPATH en `qm.ini` o en el registro y no especificando la variable de entorno.
2. Especificando MQIBindType=FASTPATH en `qm.ini` o en el registro y estableciendo la variable de entorno en FASTPATH.

Se recomienda ejecutar los escuchas como de confianza, ya que son procesos estables. Se recomienda ejecutar los canales como de confianza, a menos que esté utilizando salidas de canal inestables o el mandato STOP CHANNEL MODE(TERMINATE).

ALW Supervisión y control de canales en AIX, Linux, and Windows

Para DQM debe crear, supervisar y controlar los canales con los gestores de colas remotos. Puede controlar los canales utilizando mandatos, programas, IBM MQ Explorer, archivos para las definiciones de canal y un área de almacenamiento para la información de sincronización.

Acerca de esta tarea

Se pueden utilizar los siguientes tipos de mandato para controlar canales:

Los mandatos de IBM MQ (MQSC)
Puede utilizar los mandatos MQSC como mandatos individuales en una sesión MQSC en sistemas AIX, Linux, and Windows. Para emitir mandatos más complicados, o varios mandatos, el MQSC se puede incorporar en un archivo que luego puede ejecutarse desde la línea de mandatos. Para obtener detalles, consulte la sección Mandatos de MQSC. En este apartado se ofrecen algunos ejemplos sencillos de utilización de MQSC para la gestión de colas distribuidas.

Los mandatos de canal son un subconjunto de los mandatos de IBM MQ (MQSC). Utilice MQSC y los mandatos de control para:

- Crear, copiar, visualizar, cambiar y suprimir definiciones de canal
- Iniciar y detener canales, ejecutar mandatos ping, restablecer números de secuencia del canal y resolver los mensajes pendientes cuando no es posible restablecer los enlaces
- Mostrar información de estado sobre los canales

Mandatos de control
También puede emitir *mandatos de control* en la línea de mandatos para algunas de estas funciones. Para obtener detalles, consulte Administración de IBM MQ for Multiplatforms utilizando mandatos de control.

Mandatos de formato de mandato programable
Para obtener detalles, consulte Mandatos PCF.

Windows Linux **IBM MQ Explorer**
En sistemas Linux y Windows, se puede usar IBM MQ Explorer. Éste proporciona una interfaz de administración gráfica para realizar tareas administrativas como alternativa al uso de mandatos de control o mandatos MQSC. Las definiciones de canal se almacenan como objetos del gestor de colas.

Cada gestor de colas tiene un componente DQM para controlar las interconexiones con gestores de colas remotos compatibles. Un área de almacenamiento contiene los números de secuencia e identificadores de *unidad lógica de trabajo (LUW)*. Estos se utilizan para fines de sincronización de canal.

Para obtener una lista de las funciones disponibles al configurar y controlar los canales de mensajes, utilizando los diferentes tipos de mandato, consulte Tabla 22 en la página 264.

Procedimiento

- “Funciones necesarias para configurar y controlar canales” en la página 264
- “Iniciación a los objetos” en la página 266
- “Configuración de la comunicación en Windows” en la página 273
- “Configuración de la comunicación en AIX and Linux” en la página 281

Tareas relacionadas

“Supervisión y control de canales en IBM i” en la página 287

Utilice los mandatos y paneles de DQM para crear, supervisar y controlar los canales con gestores de colas remotos. Cada gestor de colas tiene un programa DQM para controlar las interconexiones con gestores de colas remotos compatibles.

Referencia relacionada

ALW Programas de canal en AIX, Linux, and Windows

ALW Ejemplo de planificación de canal de mensajes para AIX, Linux, and Windows
Información de configuración de ejemplo
Atributos de canal

ALW Funciones necesarias para configurar y controlar canales

Pueden ser necesarias varias funciones de IBM MQ para configurar y controlar los canales. Las funciones de canal están explicadas en este tema.

Puede crear una definición de canal utilizando los valores predeterminados suministrados por IBM MQ, especificando el nombre del canal, el tipo de canal que está creando, el método de comunicación que se utilizará, el nombre de la cola de transmisión y el nombre de la conexión.

El nombre del canal debe ser el mismo en ambos extremos del canal y exclusivo dentro de la red. Sin embargo, debe restringir los caracteres utilizados a aquellos que sean válidos para nombres de objeto de IBM MQ.

Para otras funciones relacionadas con el canal, consulte los temas siguientes:

- “Iniciación a los objetos” en la página 266
- “Crear objetos asociados” en la página 266
- “Crear objetos predeterminados” en la página 267
- “Crear un canal” en la página 267
- “Visualizar un canal” en la página 268
- “Visualización del estado del canal” en la página 268
- “Comprobación de enlaces mediante el sondeo” en la página 269
- “Iniciar un canal” en la página 269
- “Detención de un canal” en la página 271
- “Renombrar un canal” en la página 271
- “Restablecer un canal” en la página 272
- “Resolución de mensajes pendientes en un canal” en la página 272

La Tabla 22 en la página 264 muestra la lista completa de funciones de IBM MQ que podría necesitar.

Tabla 22. Funciones necesarias en sistemas AIX, Linux, and Windows

Función	**Mandatos de control**	**MQSC**	**¿Hay equivalente en IBM MQ Explorer?**
Funciones del gestor de colas			
Cambiar gestor de colas		ALTER QMGR	Sí
Crear gestor de colas	crtmqm		Sí
Suprimir gestor de colas	dltmqm		Sí
Visualizar gestor de colas		DISPLAY QMGR	Sí
Finalizar gestor de colas	endmqm		Sí
Sondear gestor de colas		PING QMGR	No
Iniciar gestor de colas	strmqm		Sí
Funciones del servidor de mandatos			
Visualizar servidor de mandatos	dspmqcsv		No
Servidor de mandatos final	endmqcsv		No

Tabla 22. Funciones necesarias en sistemas AIX, Linux, and Windows (continuación)			
Función	**Mandatos de control**	**MQSC**	**¿Hay equivalente en IBM MQ Explorer?**
Iniciar servidor de mandatos	strmqcsv		No
Funciones de cola			
Cambiar cola		ALTER QALIAS ALTER QLOCAL ALTER QMODEL ALTER QREMOTE Consulte Colas ALTER.	Sí
Borrar cola		CLEAR QLOCAL	Sí
Crear cola		DEFINE QALIAS DEFINE QLOCAL DEFINE QMODEL DEFINE QREMOTE Consulte Colas DEFINE.	Sí
Suprimir cola		DELETE QALIAS DELETE QLOCAL DELETE QMODEL DELETE QREMOTE Consulte, Colas DELETE.	Sí
Visualizar cola		DISPLAY QUEUE	Sí
Funciones de proceso			
Cambiar proceso		ALTER PROCESS	Sí
Crear proceso		DEFINE PROCESS	Sí
Suprimir proceso		Suprimir proceso	Sí
Visualizar proceso		DISPLAY PROCESS	Sí
Funciones de canal			
Cambiar canal		ALTER CHANNEL	Sí
Crear canal		DEFINE CHANNEL	Sí
Suprimir canal		Suprimir canal	Sí
Visualizar canal		DISPLAY CHANNEL	Sí
Visualizar estado de canal		DISPLAY CHSTATUS	Sí
Finalizar canal		Parar canal	Sí
Sondear canal		Sondear canal	Sí
Restablecer canal		Restablecer canal	Sí
Resolver canal		Resolver canal	Sí

Tabla 22. Funciones necesarias en sistemas AIX, Linux, and Windows (continuación)			
Función	**Mandatos de control**	**MQSC**	**¿Hay equivalente en IBM MQ Explorer?**
Ejecutar canal	runmqchl	Iniciar canal	Sí
z/OS Ejecutar iniciador de canal	runmqchi	START CHINIT	No
Ejecutar escucha[1]	runmqlsr	START LISTENER	No
Finalizar escucha	endmqlsr, solo en las plataformas siguientes: • AIX AIX • Windows Sistemas Windows		No

Nota:

1. Un escucha puede iniciarse automáticamente cuando se inicia el gestor de colas.

ALW Iniciación a los objetos

Antes de que un canal pueda iniciarse, los canales deben estar definidos y sus objetos asociados deben existir y están disponibles para su uso. Este apartado le muestra cómo hacerlo.

Utilice los mandatos de IBM MQ (MQSC) o IBM MQ Explorer para:

1. Definir canales de mensajes y objetos asociados
2. Supervisar y controlar canales de mensajes

Los objetos asociados que puede necesitar definir son:

- Colas de transmisión
- Definiciones de colas remotas
- Definiciones de alias de gestor de colas
- Definiciones de alias de colas de respuesta
- Colas locales de respuestas
- Procesos para desencadenamiento (MCA)
- Definiciones de canal de mensajes

Para poder ejecutar un canal, el enlace de comunicaciones específico para cada canal debe estar previamente definido y disponible. Para tener una descripción de cómo están definidos los enlaces LU 6.2, TCP/IP, NetBIOS, SPX y DECnet, consulte la guía de comunicación específica de la instalación. Consulte también Ejemplo de información de configuración.

Para obtener más información acerca de cómo crear y trabajar con objetos, consulte los subtemas siguientes:

ALW *Crear objetos asociados*

MQSC se utiliza para crear objetos asociados.

Utilice MQSC para crear los objetos de colas y alias: colas de transmisión, definiciones de colas remotas, definiciones de alias de gestor de colas, definiciones de alias de colas de respuesta y colas locales de respuesta.

Además, cree las definiciones de procesos para desencadenamiento (MCA) de forma similar.

Si desea ver un ejemplo sobre cómo crear todos los objetos necesarios, consulte Ejemplo de planificación de canal de mensajes para AIX, Linux, and Windows.

ALW *Crear objetos predeterminados*

Los objetos predeterminados se crean automáticamente cuando se crea un gestor de colas. Estos objetos son colas, canales, una definición de proceso y las colas de administración. Después de que se han creado los objetos predeterminados, puede sustituirlos en cualquier momento ejecutando el mandato strmqm con la opción -c.

Cuando se utiliza el mandato crtmqm para crear un gestor de colas, el mandato también inicia un programa para crear un conjunto de objetos predeterminados.

1. Los objetos predeterminados se crean uno por uno. El programa keeps mantiene un recuento de cuántos objetos se han definido correctamente, cuántos existían y se sustituyeron y cuánto intentos incorrectos se llevaron a cabo.
2. El programa le muestra los resultados y si se han producido errores, le indica que el registro de errores adecuado para obtener más detalles.

Cuando el programa finalice su ejecución, puede utilizar el mandato strmqm para iniciar el gestor de colas.

Consulte Administración de IBM MQ for Multiplatforms utilizando mandatos de control para obtener más información sobre los mandatos crtmqm y strmqm.

Modificación de los objetos predeterminados

Cuando especifica la opción -c, el gestor de colas se inicia temporalmente mientras los objetos se crean y luego se cierra de nuevo. Si se emite strmqm con la opción -c, se renuevan los objetos del sistema existentes con los valores predeterminados (por ejemplo, el atributo MCAUSER de una definición de canal se establece en espacios en blanco). Debe utilizar el mandato strmqm de nuevo, sin la opción -c, si desea iniciar el gestor de colas.

Si desea cambiar los objetos predeterminados, puede crear su propia versión del antiguo archivo amqscoma.tst y editarlo.

ALW *Crear un canal*

Cree dos definiciones de canal, una en cada extremo de la conexión. La primera definición de canal se crea en el primer gestor de colas. A continuación, se crea la segunda definición de canal en el segundo gestor de colas, en el otro extremo del enlace.

Ambos extremos tienen que definirse con el mismo nombre de canal. Ambos extremos han de tener tipos de canal compatibles, por ejemplo: emisor y receptor.

Para crear una definición de canal para un extremo del enlace utilice el mandato de MQSC DEFINE CHANNEL. Incluya el nombre del canal, el tipo de canal para este extremo de la conexión, un nombre de conexión, una descripción (si es necesario), el nombre de la cola de transmisión (si es necesario) y el protocolo de transmisión. Además, incluya cualquier otro atributo que desee que sea distinto de los valores predeterminados del sistema para el tipo de canal necesario, utilizando la información que ha recopilado anteriormente.

En la sección Atributos de canal se le ofrece ayuda para decidir los valores de los atributos del canal.

Nota: Es recomendable dar un nombre exclusivo a todos los canales de la red. Incluir los nombres de los gestores de colas de origen y destino en el nombre del canal es una buena forma de hacerlo.

Ejemplo de creación de canal

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) +
DESCR('Sender channel to QM2') +
CONNAME(QM2) TRPTYPE(TCP) XMITQ(QM2) CONVERT(YES)
```

En todos los ejemplos de MQSC, el mandato se muestra tal como aparece en un archivo de mandatos y tal como se escribe en AIX, Linux, and Windows. Los dos métodos parecen idénticos, excepto que para emitir un mandato de forma interactiva, primero debe iniciar una sesión de MQSC. Escriba `runmqsc`, para el gestor de colas predeterminado, o `runmqsc` *qmname* donde *qmname* es el nombre del gestor de colas necesario. A continuación, escriba cualquier número de mandatos, tal como se muestra en los ejemplos.

Para una mayor portabilidad, restrinja la longitud de la línea de los mandatos a 72 caracteres. Utilice el carácter de concatenación, +, tal como se muestra para continuar en más de una línea:

- Windows En Windows, utilice Control-z para finalizar la entrada en la línea de mandatos.
- Linux AIX En AIX and Linux, use Ctrl-d.
- Como alternativa, en AIX, Linux, and Windows, utilice el mandato **end**.

ALW *Visualizar un canal*

Utilice el mandato MQSC DISPLAY CHANNEL para visualizar los atributos de un canal.

El parámetro ALL del mandato DISPLAY CHANNEL se emite de forma predeterminada si no se solicita ningún atributo específico y el nombre de canal especificado no es genérico.

Los atributos se describen en la sección Atributos de canal.

Ejemplos de visualización de canal

```
DISPLAY CHANNEL(QM1.TO.QM2) TRPTYPE,CONVERT
DISPLAY CHANNEL(QM1.TO.*) TRPTYPE,CONVERT
DISPLAY CHANNEL(*) TRPTYPE,CONVERT
DISPLAY CHANNEL(QM1.TO.QMR34) ALL
```

ALW *Visualización del estado del canal*

Utilice el mandato MQSC DISPLAY CHSTATUS, especificando el nombre del canal y si desea el estado actual de los canales o el estado de la información guardada.

DISPLAY CHSTATUS se aplica a todos los canales de mensajes. No se aplica a los canales MQI que no sean canales de conexión con el servidor.

La información visualizada incluye:

- Nombre de canal
- Nombre de la conexión de comunicaciones
- Estado de pendiente del canal (cuando sea apropiado)
- Último número de secuencia
- Nombre de cola de transmisión (cuando sea apropiado)
- Identificador pendiente (cuando sea adecuado)
- Último número de secuencia confirmado
- Identificador de unidad lógica de trabajo
- ID de proceso
- Windows ID de hebra (sólo Windows)

Ejemplos de visualización de canal

```
DISPLAY CHSTATUS(*) CURRENT
```

```
DISPLAY CHSTATUS(QM1.TO.*) SAVED
```

El estado guardado no se aplica hasta que se haya transmitido como mínimo un lote de mensajes en el canal. El estado también se guarda cuando se detiene un canal (mediante el mandato STOP CHL) y cuando finaliza un gestor de colas.

ALW *Comprobación de enlaces mediante el sondeo*

Utilice el mandato MQSC **PING CHANNEL** para intercambiar un mensaje de datos fijo con el extremo remoto.

El sondeo da cierta confianza al supervisor del sistema de que el enlace está disponible y en funcionamiento.

Ping no implica el uso de colas de transmisión y colas de destino. Utiliza definiciones de canal, el enlace de comunicaciones relacionado y la configuración de la red. Sólo se puede utilizar si el canal no está activo actualmente.

Está disponible únicamente desde los canales emisores, servidores y de clúster emisor. El canal correspondiente se inicia en el extremo del enlace y realiza la negociación del parámetro de inicio. Los errores se notifican con normalidad.

El resultado del intercambio de mensajes se presenta como `Ping complete` o mediante un mensaje de error.

Sondeo con LU 6.2

Cuando se invoca Ping, de forma predeterminada ningún ID de usuario ni ninguna contraseña fluyen al extremo receptor. Si se necesita un ID de usuario y una contraseña, pueden crearse en el extremo de inicio en la definición de canal. Si se especifica una contraseña en la definición de canal, IBM MQ la cifra antes de guardarla. A continuación se descifra antes de que fluya a través de la conversación.

Tareas relacionadas

Utilización de ping para probar las comunicaciones

Hacer ping a un canal para verificar una conexión

Referencia relacionada

PING CHANNEL (probar respuesta de canal)

ALW *Iniciar un canal*

Utilice el mandato MQSC START CHANNEL para canales emisor, servidor y peticionario. Para que las aplicaciones puedan intercambiar mensajes, debe iniciar un programa de escucha para conexiones de entrada.

START CHANNEL no es necesario en el caso de que se haya configurado un canal con desencadenamiento de gestor de colas.

Cuando se inicia, el MCA emisor lee las definiciones de canal y abre la cola de transmisión. Se emite una secuencia de inicio de canal, que inicia de forma remota el MCA correspondiente del canal receptor o servidor. Cuando se han iniciado, los procesos del emisor y servidor esperan hasta que llegan mensajes a la cola de transmisión y los transmiten cuando lleguen.

Cuando se utiliza el desencadenamiento o se ejecutan canales como hebras, asegúrese de que el iniciador de canal está disponible para supervisar la cola de inicio. El iniciador de canal se inicia de forma predeterminada como parte del gestor de colas.

Sin embargo, TCP y LU 6.2 proporcionan otras posibilidades:

- Linux AIX En el caso de TCP en AIX and Linux, inetd puede configurarse para iniciar un canal. inetd se inicia como un proceso separado.
- Linux AIX En el caso de LU 6.2 en sistemas AIX and Linux, configure el producto SNA para iniciar el proceso de respuesta de LU 6.2.

- Windows En el caso de LU 6.2 en Windows, el uso de SNA Server puede utilizar TpStart (una utilidad que suministra SNA Server) para iniciar un canal. TpStart se inicia como un proceso separado.

El uso de la opción Iniciar hace que el canal se resincronice cuando sea necesario.

Para que el inicio se realice correctamente:

- Las definiciones de canal, locales y remotas, deben existir. Si no hay ninguna definición de canal adecuada para un canal receptor o de conexión con el servidor, se crea una predeterminada automáticamente si el canal está definido automáticamente. Consulte Programa de salida de definición automática de canal.
- La cola de transmisión debe existir y no tener ningún otro canal que la utilice.
- Los MCA, locales y remotos, deben existir.
- El enlace de comunicaciones debe estar disponible.
- Los gestores de colas debe estar en ejecución, locales y remotos.
- El canal de mensajes no deben estar todavía en ejecución.

Se devuelve un mensaje a la pantalla que confirma que la solicitud para iniciar un canal se ha aceptado. Para confirmar que el mandato de inicio se ha ejecutado correctamente, compruebe el registro de errores o utilice DISPLAY CHSTATUS. Los registros de errores son:

Windows **Windows**

MQ_DATA_PATH\qmgrs\qmname\errors\AMQERR01.LOG (para cada gestor de colas denominado qmname)

MQ_DATA_PATH\qmgrs\@SYSTEM\errors\AMQERR01.LOG (para errores generales)

MQ_DATA_PATH representa el directorio de alto nivel en el que está instalado IBM MQ .

Nota: En Windows, también se sigue obteniendo un mensaje en el registro de sucesos de aplicación de sistemas Windows.

Linux AIX **AIX and Linux**

/var/mqm/qmgrs/qmname/errors/AMQERR01.LOG (para cada gestor de colas denominado qmname)

/var/mqm/qmgrs/@SYSTEM/errors/AMQERR01.LOG (para errores generales)

En AIX, Linux, and Windows, utilice el mandato **runmqlsr** para iniciar el proceso de escucha IBM MQ. De forma predeterminada, todas las solicitudes de entrada para conexión de canal hacen que el proceso de escucha inicie los MCA como hebras del proceso amqrmppa.

```
runmqlsr -t tcp -m QM2
```

Para conexiones de salida, debe iniciar el canal de las tres maneras siguientes:

1. Utilice el mandato de MQSC START CHANNEL, que especifica el nombre del canal, para iniciar el canal como un proceso o hebra, en función del parámetro MCATYPE. (Si los canales se inician como hebras, son hebras de un iniciador de canal.)

   ```
   START CHANNEL(QM1.TO.QM2)
   ```

2. Utilice el mandato de control runmqchl para iniciar el canal como un proceso.

   ```
   runmqchl -c QM1.TO.QM2 -m QM1
   ```

3. Utilice el iniciador de canal para desencadenar el canal.

ALW *Detención de un canal*

Utilice el mandato de MQSC STOP CHANNEL para solicitar que el canal detenga la actividad. El canal no inicia un nuevo lote de mensajes hasta que el operador inicia el canal de nuevo.

Para obtener información sobre el reinicio de canales detenidos, consulte "Reinicio de canales detenidos" en la página 251.

Este mandato se puede emitir en un canal de cualquier tipo excepto MQCHT_CLNTCONN.

Puede seleccionar el tipo de detención que necesite:

Ejemplo de detención con inmovilización

```
STOP CHANNEL(QM1.TO.QM2) MODE(QUIESCE)
```

Este mandato solicita al canal el cierre de forma ordenada. El lote actual de mensajes ha finalizado y se realiza el procedimiento del punto de sincronización con el otro extremo del canal. Si el canal está desocupado este mandato no termina un canal receptor.

Ejemplo de detención forzosa

```
STOP CHANNEL(QM1.TO.QM2) MODE(FORCE)
```

Esta opción detiene el canal de inmediato, pero no termina el proceso o hebra del canal. El canal no completa el proceso del lote actual de mensajes y puede, por consiguiente, dejar el canal pendiente. En general, considere utilizar la opción de detención con desactivación temporal.

Ejemplo de detención de terminación

```
STOP CHANNEL(QM1.TO.QM2) MODE(TERMINATE)
```

Esta opción detiene el canal de inmediato y termina el proceso o hebra del canal.

Ejemplo de detención (inmovilización) detenida

```
STOP CHANNEL(QM1.TO.QM2) STATUS(STOPPED)
```

Este mandato no especifica MODE; por consiguiente, toma el valor predeterminado de MODE(QUIESCE). Solicita que el canal se detenga de modo que no se pueda reiniciar automáticamente, pero debe iniciarse manualmente.

Ejemplo de detención (inmovilización) inactiva

```
STOP CHANNEL(QM1.TO.QM2) STATUS(INACTIVE)
```

Este mandato no especifica MODE; por consiguiente, toma el valor predeterminado de MODE(QUIESCE). Solicita que el canal esté inactivo de forma que se reinicie automáticamente cuando sea necesario.

ALW *Renombrar un canal*

Utilice MQSC para renombrar un canal de mensajes.

Utilice MQSC para efectuar los pasos siguientes:

1. Utilice STOP CHANNEL para detener el canal.
2. Utilice DEFINE CHANNEL para crear una definición de canal duplicada con el nombre nuevo.

3. Utilice DISPLAY CHANNEL para comprobar si se ha creado correctamente.
4. Utilice DELETE CHANNEL para suprimir la definición de canal original.

Si decide renombrar un canal de mensajes, no olvide que un canal tiene dos definiciones de canal, una en cada extremo. Asegúrese de renombrar el canal en ambos extremos al mismo tiempo.

ALW *Restablecer un canal*

Utilice el mandato de MQSC RESET CHANNEL para cambiar el número de secuencia de mensaje.

El mandato RESET CHANNEL está disponible para cualquier canal de mensajes, pero no para canales MQI (de conexión con el cliente o de conexión con el servidor). El primer mensaje inicia la nueva secuencia la próxima vez que se inicia el canal.

Si el mandato se emite en un canal de remitente o de servidor, informa al otro lado del cambio cuando el canal se reinicia.

Conceptos relacionados

"Iniciación a los objetos" en la página 266
Antes de que un canal pueda iniciarse, los canales deben estar definidos y sus objetos asociados deben existir y están disponibles para su uso. Este apartado le muestra cómo hacerlo.

"Función de control de canales" en la página 240
La función de control de canales proporciona recursos para definir, supervisar y controlar canales.

Tareas relacionadas

"Configuración de la gestión de colas distribuidas" en la página 210
En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

Referencia relacionada

RESET CHANNEL

ALW *Resolución de mensajes pendientes en un canal*

Utilice el mandato MQSC RESOLVE CHANNEL cuando un emisor o servidor mantiene los mensajes pendientes. Por ejemplo, porque un extremo del enlace ha terminado y no hay ninguna perspectiva de recuperación.

El mandato RESOLVE CHANNEL acepta uno de los dos parámetros: BACKOUT o COMMIT. La restitución restaura mensajes a la cola de transmisión, mientras que la confirmación los descarta.

El programa del canal no intenta establecer una sesión con un socio. En su lugar, determina el identificador de unidad lógica de trabajo (LUWID) que representa los mensajes pendientes. A continuación, emite, tal como se solicitó:

- BACKOUT para restaurar los mensajes a la cola de transmisión; o
- COMMIT para suprimir los mensajes de la cola de transmisión.

Para que la resolución se realice correctamente:

- El canal debe estar inactivo
- El canal debe estar pendiente
- El tipo de canal debe ser emisor, servidor o remitente de clúster
- Debe existir una definición de canal local
- El gestor de colas local debe estar en ejecución

Conceptos relacionados

"Iniciación a los objetos" en la página 266
Antes de que un canal pueda iniciarse, los canales deben estar definidos y sus objetos asociados deben existir y están disponibles para su uso. Este apartado le muestra cómo hacerlo.

"Función de control de canales" en la página 240
La función de control de canales proporciona recursos para definir, supervisar y controlar canales.

Tareas relacionadas

"Configuración de la gestión de colas distribuidas" en la página 210
En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

Referencia relacionada

RESOLVE CHANNEL

Windows Configuración de la comunicación en Windows

Cuando se inicia un canal de gestión de colas distribuidas, éste intenta utilizar la conexión especificada en la definición de canal. Para que esto tenga éxito, la conexión debe estar definida y disponible. En esta sección se explica cómo hacerlo empleando las formas de comunicación disponibles en sistemas IBM MQ for Windows.

Antes de empezar

Puede que le resulte útil consultar Configuración de ejemplo - IBM MQ for Windows.

MQ Adv. CD Un canal de mensajes que utiliza TCP/IP puede apuntar a un IBM Aspera faspio Gateway, que proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de la red. Un gestor de colas que se ejecuta en cualquier plataforma autorizada puede conectarse a través de un Aspera gateway. La propia pasarela se despliega en Red Hat o Ubuntu Linux, o Windows. Consulte Definición de una conexión de Aspera gateway en Linux o Windows.

Acerca de esta tarea

Al configurar la comunicación para IBM MQ en Windows, se puede elegir entre los siguientes tipos de comunicación:

- TCP/IP
- LU6.2
- NetBIOS

Procedimiento

- Para obtener información sobre cómo configurar la comunicación en un sistema Windows, consulte el subtema del tipo de comunicación elegido:
  - "Definición de una conexión TCP en Windows" en la página 274
  - "Definición de una conexión LU 6.2 en Windows" en la página 276
  - "Definición de una conexión NetBIOS en Windows" en la página 278

  No todas las funciones e instalaciones de IBM MQ for Windows están disponibles en entornos que utilizan protocolos de comunicaciones que no sean TCP/IP. El elemento que no está disponible es IBM MQ Explorer.

Tareas relacionadas

"Supervisión y control de canales en AIX, Linux, and Windows" en la página 263
Para DQM debe crear, supervisar y controlar los canales con los gestores de colas remotos. Puede controlar los canales utilizando mandatos, programas, IBM MQ Explorer, archivos para las definiciones de canal y un área de almacenamiento para la información de sincronización.

"Configuración de conexiones entre el cliente y el servidor" en la página 16

Para configurar los enlaces de comunicación entre IBM MQ MQI clients y servidores, decida el protocolo de comunicación, defina las conexiones en ambos extremos del enlace, inicie un escucha y defina canales.

"Configuración de la comunicación en AIX and Linux" en la página 281
Cuando se inicia un canal de gestión de colas distribuidas, éste intenta utilizar la conexión especificada en la definición de canal. Para que esto tenga éxito, la conexión debe estar definida y disponible. En esta sección se explica cómo hacerlo empleando las formas de comunicación disponibles en sistemas IBM MQ for UNIX or Linux.

Referencia relacionada

"Qué tipo de comunicación utilizar" en la página 17
Diferentes plataformas dan soporte a diferentes protocolos de comunicación. El protocolo de transmisión que elija dependerá de su combinación de plataformas de servidor y IBM MQ MQI client.

Windows *Definición de una conexión TCP en Windows*

Defina una conexión TCP configurando un canal en el extremo emisor para especificar la dirección del destino y ejecutando un programa de escucha en el extremo receptor.

Antes de empezar

MQ Adv. CD Un canal de mensajes que utiliza TCP/IP puede apuntar a un IBM Aspera faspio Gateway, que proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de la red. Un gestor de colas que se ejecuta en cualquier plataforma autorizada puede conectarse a través de un Aspera gateway. La propia pasarela se despliega en Red Hat o Ubuntu Linux, o Windows. Consulte Definición de una conexión de Aspera gateway en Linux o Windows.

Extremo emisor

Especifique el nombre de host o la dirección TCP de la máquina de destino, en el campo Nombre de conexión de la definición de canal.

El puerto de conexión toma el valor predeterminado 1414. El número de puerto 1414 es el asignado por la IANA (Internet Assigned Numbers Authority) para IBM MQ.

Para utilizar un número de puerto distinto al valor predeterminado, especifique el campo de nombre de conexión de la definición de objeto de canal de este modo:

```
DEFINE CHANNEL('channel name') CHLTYPE(SDR) +
          TRPTYPE(TCP) +
          CONNAME('OS2ROG3(1822)') +
          XMITQ('XMitQ name') +
          REPLACE
```

donde 0S2R0G3 es el nombre DNS del gestor de colas remoto y 1822 es el puerto necesario. (Debe ser el puerto en el que el escucha del extremo receptor está a la escucha).

Un canal en ejecución debe detenerse y reiniciarse para captar cualquier cambio en la definición de objeto de canal.

Puede cambiar el número de puerto predeterminado especificándolo en el archivo `.ini` para IBM MQ for Windows:

```
TCP:
Port=1822
```

Nota: Para seleccionar qué número de puerto TCP/IP utilizar, IBM MQ utiliza el primer número de puerto que encuentra en la secuencia siguiente:

1. El número de puerto especificado explícitamente en la definición de canal o en la línea de mandatos. Este número permite que un canal altere temporalmente el número de puerto predeterminado.

2. El atributo del puerto especificado en la stanza TCP del archivo `.ini`. Este número permite que un gestor de colas altere temporalmente el número de puerto predeterminado.
3. El valor predeterminado de 1414. Este es el número asignado a IBM MQ por IANA (Internet Assigned Numbers Authority) para conexiones entrantes y salientes.

Si desea más información sobre los valores que puede definir utilizando qm.ini, consulte Stanzas del archivo de configuración para la gestión de colas distribuida.

Recepción en TCP

Para iniciar un programa de canal receptor, debe haber iniciado un programa de escucha que detecte las solicitudes de red entrantes e iniciar el canal asociado. Puede utilizar el escucha de IBM MQ.

Los programas de canal receptor se inician en respuesta a una solicitud de inicio del canal emisor.

Para iniciar un programa de canal receptor, debe haber iniciado un programa de escucha que detecte las solicitudes de red entrantes e iniciar el canal asociado. Puede utilizar el escucha de IBM MQ.

Para ejecutar el escucha proporcionado con IBM MQ, que inicia nuevos canales como hebras, utilice el mandato runmqlsr.

Un ejemplo básico de la utilización del mandato **runmqlsr**:

```
runmqlsr -t tcp [-m QMNAME] [-p 1822]
```

Los corchetes indican parámetros opcionales; QMNAME no es necesario para el gestor de colas predeterminado y el número de puerto no es necesario si está utilizando el valor predeterminado (1414). El número de puerto no debe exceder 65535.

Nota: Para seleccionar qué número de puerto TCP/IP utilizar, IBM MQ utiliza el primer número de puerto que encuentra en la secuencia siguiente:

1. El número de puerto especificado explícitamente en la definición de canal o en la línea de mandatos. Este número permite que un canal altere temporalmente el número de puerto predeterminado.
2. El atributo del puerto especificado en la stanza TCP del archivo `.ini`. Este número permite que un gestor de colas altere temporalmente el número de puerto predeterminado.
3. El valor predeterminado de 1414. Este es el número asignado a IBM MQ por IANA (Internet Assigned Numbers Authority) para conexiones entrantes y salientes.

Para obtener un rendimiento óptimo, ejecute el escucha de IBM MQ como una aplicación de confianza, como se describe en “Ejecución de canales y escuchas como aplicaciones de confianza” en la página 262. Consulte Restricciones para aplicaciones de confianza para obtener información sobre aplicaciones de confianza.

Utilización de la opción TCP/IP SO_KEEPALIVE

Si desea utilizar la opción SO_KEEPALIVE de Windows debe añadir la siguiente entrada en el registro:

```
TCP:
KeepAlive=yes
```

Para obtener más información sobre la opción SO_KEEPALIVE, consulte “Cómo comprobar que el otro extremo del canal sigue estando disponible” en la página 247.

En Windows, el valor de registro `HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters` para la opción Windows **KeepAliveTime** controla el intervalo que transcurre antes de que se compruebe la conexión. El valor predeterminado es dos horas.

Utilización de la opción de reserva de escucha TCP

En TCP, las conexiones se tratan de forma incompleta a menos que tenga lugar un reconocimiento entre el servidor y el cliente. Estas conexiones se llaman solicitudes de conexión pendientes. Se establece un valor máximo para estas solicitudes de conexión pendientes y se puede considerar una reserva de solicitudes en espera del puerto TCP para que el escucha acepte la solicitud.

Consulte "Utilización de la opción de proceso de escucha TCP en IBM MQ for Multiplatforms" en la página 284 para obtener más información y el valor específico para Windows.

Windows *Definición de una conexión LU 6.2 en Windows*

SNA debe configurarse de manera que pueda establecerse una conversación LU 6.2 entre las dos máquinas.

Una vez que configurado el SNA, haga lo siguiente.

Consulte la tabla siguiente para obtener más información.

Tabla 23. Valores del sistema Windows local para una plataforma de gestor de colas remoto

Plataforma remota	**TPNAME**	**TPPATH**
z/OS o MVS/ESA sin CICS	El mismo que el de la información complementaria correspondiente sobre el gestor de colas remoto.	-
z/OS o MVS/ESA utilizando CICS	CRCK (emisor) CKSV (peticionario) CRCK (servidor)	-
IBM i	El mismo que el valor de comparación de la entrada de direccionamiento del sistema IBM i.	-
Sistemas AIX and Linux	El mismo que el de la información complementaria correspondiente sobre el gestor de colas remoto.	*MQ_INSTALLATION_PATH*/bin/amqcrs6a
Windows	El mismo que el especificado en el mandato Run Listener de Windows, o el programa de transacción invocable definido mediante TpSetup en Windows.	*MQ_INSTALLATION_PATH*\bin\amqcrs6a

MQ_INSTALLATION_PATH representa el directorio de alto nivel en el que está instalado IBM MQ.

Si tiene más de un gestor de colas en la misma máquina, asegúrese de que los TPnames de las definiciones de canal son exclusivos.

Para obtener la información más reciente sobre la configuración de AnyNet SNA sobre TCP/IP, consulte la siguiente documentación en línea de IBM: AnyNet SNA sobre TCP/IP y Operaciones de nodo SNA.

Conceptos relacionados

"Extremo emisor en LU 6.2 en Windows" en la página 277
Cree un objeto complementario CPI-C (destino simbólico) en la aplicación de administración del producto LU 6.2 que está utilizando. Especifique este nombre en el campo Nombre de conexión de la definición de canal. Cree también un enlace LU 6.2 al socio.

"Recepción en LU 6.2 en Windows" en la página 277
Los programas de canal receptor se inician en respuesta a una solicitud de inicio del canal emisor.

Windows *Extremo emisor en LU 6.2 en Windows*

Cree un objeto complementario CPI-C (destino simbólico) en la aplicación de administración del producto LU 6.2 que está utilizando. Especifique este nombre en el campo Nombre de conexión de la definición de canal. Cree también un enlace LU 6.2 al socio.

En el objeto del lado de CPI-C entre el nombre de LU asociado en la máquina receptora, el nombre de TP y el nombre de modalidad. Por ejemplo:

```
Partner LU Name              OS2ROG2
Partner TP Name              recv
Mode Name                    #INTER
```

Windows *Recepción en LU 6.2 en Windows*

Los programas de canal receptor se inician en respuesta a una solicitud de inicio del canal emisor.

Para iniciar un programa de canal receptor, es preciso iniciar un programa de escucha para detectar solicitudes de red entrantes e iniciar el canal asociado. Este programa de escucha se inicia con el mandato RUNMQLSR, indicando el TpName en el que realizar la escucha. O bien, puede utilizar TpStart bajo SNA Server para Windows.

Utilización del mandato RUNMQLSR

Ejemplo del mandato para iniciar el escucha:

```
RUNMQLSR -t LU62 -n RECV -m QMNAME
```

Donde RECV es el TpName que se ha especificado en el otro extremo (emisor) como "TpName para iniciar en el extremo remoto". El parámetro **-m** utilizado en la última parte de este mandato es opcional y no es necesario para el gestor de colas predeterminado.

Es posible tener más de un gestor de colas en ejecución en una máquina. Debe asignar un TpName diferente para cada gestor de colas y, a continuación, iniciar un programa de escucha para cada uno. Por ejemplo:

```
RUNMQLSR -t LU62 -m QM1 -n TpName1
RUNMQLSR -t LU62 -m QM2 -n TpName2
```

Para obtener un rendimiento óptimo, ejecute el escucha de IBM MQ como una aplicación de confianza, como se describe en Ejecución de canales y escuchas como aplicaciones de confianza. Consulte Restricciones para aplicaciones de confianza para obtener información sobre aplicaciones de confianza.

Puede detener todos los escuchas de IBM MQ que se ejecutan en un gestor de colas que está inactivo mediante el mandato:

```
ENDMQLSR -m QMNAME
```

Utilización de Microsoft SNA Server en Windows

Puede utilizar TpSetup (desde SNA Server SDK) para definir un TP invocable que a continuación, activa amqcrs6a.exe, o bien puede establecer varios valores de registro manualmente. Los parámetros que se deben pasar a amqcrs6a.exe son:

```
-m QM -n TpName
```

donde *QM* es el nombre del gestor de colas y *TpName* es el nombre TP. Para obtener más información, consulte la publicación *Microsoft SNA Server APPC Programmers Guide* o *Microsoft SNA Server CPI-C Programmers Guide*.

Si no especifica un nombre de gestor de colas, se toma el gestor de colas predeterminado.

Windows *Definición de una conexión NetBIOS en Windows*

Una conexión NetBIOS se aplica únicamente a un cliente y un servidor que ejecuten Windows. IBM MQ utiliza tres tipos de recursos NetBIOS al establecer una conexión NetBIOS con otro producto IBM MQ: sesiones, mandatos y nombres. Cada uno de estos recursos tiene un límite, que se establece ya sea de forma predeterminada o por elección propia durante la instalación de NetBIOS.

Cada canal en ejecución, independientemente del tipo, utiliza una sesión NetBIOS y un mandato NetBIOS. La implementación NetBIOS de IBM permite que varios procesos utilicen el mismo nombre NetBIOS local. Por lo tanto, sólo es necesario que haya un nombre NETBIOS disponible para que lo utilice IBM MQ. Las implementaciones de otros proveedores como, por ejemplo, la emulación NetBIOS de Novell, requieren un nombre local diferente para cada proceso. Compruebe sus requisitos a partir de la documentación del producto NetBIOS que está utilizando.

En cualquier caso, asegúrese de que dispone de recursos suficientes de cada tipo o aumente el máximo especificado en la configuración. Los cambios en los valores requieren reiniciar el sistema.

Durante el arranque del sistema, el controlador de dispositivo NetBIOS muestra el número de sesiones, los mandatos y los nombres disponibles para que las aplicaciones. Estos recursos están disponibles para todas las aplicaciones basadas en NetBIOS que se ejecuten en el mismo sistema. Por consiguiente, es posible que otras aplicaciones consuman estos recursos antes de que IBM MQ necesite adquirirlos. El administrador de red debe aclarar este asunto.

Conceptos relacionados

“Definición del nombre NETBIOS local de IBM MQ” en la página 278
El nombre NetBIOS local que los procesos de canal de IBM MQ utilizan se puede especificar de tres modos.

“Establecer los límites de nombres, mandatos y sesiones NetBIOS del gestor de colas” en la página 279
Los límites del gestor de colas para las sesiones NetBIOS, los mandatos y los nombres pueden especificarse de dos maneras.

“Establecer el número de adaptador LAN” en la página 279
Para que los canales funcionen correctamente con NetBIOS, el soporte del adaptador en cada extremo debe ser compatible. IBM MQ le permite controlar la elección del número del adaptador LAN (LANA) utilizando el valor AdapterNum en la stanza NETBIOS del archivo qm.ini especificando el parámetro **-a** en el mandato runmqlsr.

“Inicio de la conexión NetBIOS” en la página 280
Definición de los pasos necesarios para iniciar una conexión.

“Definición del escucha de destino de la conexión NetBIOS” en la página 280
Definición de los pasos que se deben realizar en el extremo receptor de la conexión NetBIOS.

Windows *Definición del nombre NETBIOS local de IBM MQ*

El nombre NetBIOS local que los procesos de canal de IBM MQ utilizan se puede especificar de tres modos.

Por orden de prioridad las tres maneras son:

1. El valor especificado en el parámetro **-l** del mandato **runmqlsr** , por ejemplo:

   ```
   runmqlsr -t netbios -l my_station
   ```

2. La variable de entorno **MQNAME** con un valor establecido por el mandato:

   ```
   SET MQNAME= my_station
   ```

   Por ejemplo:

```
SET MQNAME=CLIENT1
```

Puede establecer el valor **MQNAME** para cada proceso. De forma alternativa, puede establecerlo a nivel de sistema en el registro de Windows.

Si está utilizando una implementación de NetBIOS que requiere nombres exclusivos, debe emitir un mandato **SET MQNAME** en cada ventana en la que se inicie un proceso de IBM MQ . El valor **MQNAME** es arbitrario, pero debe ser exclusivo para cada proceso.

3. La stanza NETBIOS en el archivo de configuración del gestor de colas `qm.ini`. Por ejemplo:

```
NETBIOS:

LocalName= my_station
```

Nota:

1. Debido a las variaciones en la implementación de los productos de NetBIOS soportados, es aconsejable que cada nombre de NetBIOS sea exclusivo en la red. De lo contrario, pueden producirse resultados imprevisibles. Si tiene problemas para establecer un canal NetBIOS y hay mensajes de error en el registro de errores del gestor de colas que muestran un código de retorno NetBIOS de X'15', revise su uso de nombres NetBIOS.
2. En Windows, no puede utilizar el nombre de la máquina como nombre de NetBIOS porque ya lo utiliza Windows.
3. El inicio del canal emisor requiere que se especifique un nombre NetBIOS utilizando la variable de entorno MQNAME o LocalName en el archivo qm.ini.

Windows *Establecer los límites de nombres, mandatos y sesiones NetBIOS del gestor de colas*

Los límites del gestor de colas para las sesiones NetBIOS, los mandatos y los nombres pueden especificarse de dos maneras.

Por orden de prioridad existen las maneras siguientes:

1. Los valores especificados en el mandato RUNMQLSR:

```
-s Sessions
-e Names
-o Commands
```

Si el operando -m no se especifica en el mandato, los valores sólo se aplican al gestor de colas predeterminado.

2. La stanza NETBIOS en el archivo de configuración del gestor de colas qm.ini. Por ejemplo:

```
NETBIOS:

NumSess= Qmgr_max_sess
NumCmds= Qmgr_max_cmds
NumNames= Qmgr_max_names
```

Windows *Establecer el número de adaptador LAN*

Para que los canales funcionen correctamente con NetBIOS, el soporte del adaptador en cada extremo debe ser compatible. IBM MQ le permite controlar la elección del número del adaptador LAN (LANA) utilizando el valor AdapterNum en la stanza NETBIOS del archivo qm.ini especificando el parámetro **-a** en el mandato runmqlsr.

El número de adaptador LAN predeterminado que utiliza IBM MQ para las conexiones NetBIOS es 0. Verifique el número que se utiliza en el sistema de la forma siguiente:

En Windows, no es posible consultar el número de adaptador LAN directamente a través del sistema operativo. En su lugar, utilice el programa de utilidad de línea de mandatos LANACFG.EXE, que está

disponible en Microsoft. La salida de la herramienta muestra los números de adaptador LAN virtuales y sus enlaces efectivos. Para obtener más información sobre los números de adaptador LAN, consulte el artículo de Microsoft Knowledge Base 138037 *CÓMO: Utilizar números de LANA en un entorno de 32 bits*.

Especifique el valor correcto en la stanza NETBIOS del archivo de configuración de gestor de colas, qm.ini:

```
NETBIOS:
AdapterNum= n
```

donde n es el número del adaptador LAN para este sistema.

Windows *Inicio de la conexión NetBIOS*

Definición de los pasos necesarios para iniciar una conexión.

Para iniciar la conexión, siga estos pasos en el extremo emisor:

1. Defina el nombre de la estación de NetBIOS utilizando el valor MQNAME o LocalName.
2. Verifique el número del adaptador LAN que se utiliza en el sistema y especifique el archivo correcto utilizando el AdapterNum.
3. En el campo ConnectionName de la definición de canal, especifique el nombre de NetBIOS utilizado por el programa de escucha de destino. En Windows, los canales NetBIOS se deben ejecutar como hebras. Haga esto especificando MCATYPE(THREAD) en la definición de canal.

   ```
   DEFINE CHANNEL (chname) CHLTYPE(SDR) +
   TRPTYPE(NETBIOS) +
   CONNAME(your_station) +
   XMITQ(xmitq) +
   MCATYPE(THREAD) +
   REPLACE
   ```

Windows *Definición del escucha de destino de la conexión NetBIOS*

Definición de los pasos que se deben realizar en el extremo receptor de la conexión NetBIOS.

En el extremo receptor, siga estos pasos:

1. Defina el nombre de la estación de NetBIOS utilizando el valor MQNAME o LocalName.
2. Verifique el número del adaptador LAN que se utiliza en el sistema y especifique el archivo correcto utilizando el AdapterNum.
3. Defina el canal receptor:

   ```
   DEFINE CHANNEL (chname) CHLTYPE(RCVR) +
   TRPTYPE(NETBIOS) +
   REPLACE
   ```

4. Inicie el programa de escucha de IBM MQ para establecer la estación y que sea posible ponerse en contacto con ella. Por ejemplo:

   ```
   RUNMQLSR -t NETBIOS -l your_station [-m qmgr]
   ```

   Este mandato establece `your_station`como una estación de NetBIOS a la espera de ser contactada. El nombre de la estación de NetBIOS debe ser exclusivo en toda la red NetBIOS.

Para obtener un rendimiento óptimo, ejecute el escucha de IBM MQ como una aplicación de confianza, como se describe en “Ejecución de canales y escuchas como aplicaciones de confianza” en la página 262. Consulte Restricciones para aplicaciones de confianza para obtener información sobre aplicaciones de confianza.

Puede detener todos los escuchas de IBM MQ que se ejecutan en un gestor de colas que está inactivo mediante el mandato:

```
ENDMQLSR [-m QMNAME]
```

Si no especifica un nombre de gestor de colas, se toma el gestor de colas predeterminado.

Linux AIX Configuración de la comunicación en AIX and Linux

Cuando se inicia un canal de gestión de colas distribuidas, éste intenta utilizar la conexión especificada en la definición de canal. Para que esto tenga éxito, la conexión debe estar definida y disponible. En esta sección se explica cómo hacerlo empleando las formas de comunicación disponibles en sistemas IBM MQ for UNIX or Linux.

Antes de empezar

Puede que le resulte útil consultar los apartados siguientes:

- AIX Configuración de ejemplo - IBM MQ for AIX
- Linux Configuración de ejemplo - IBM MQ para Linux

MQ Adv. CD Un canal de mensajes que utiliza TCP/IP puede apuntar a un IBM Aspera faspio Gateway, que proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de la red. Un gestor de colas que se ejecuta en cualquier plataforma autorizada puede conectarse a través de un Aspera gateway. La propia pasarela se despliega en Red Hat o Ubuntu Linux, o Windows. Consulte Definición de una conexión de Aspera gateway en Linux o Windows.

Acerca de esta tarea

Cuando se inicia un canal de gestión de colas distribuidas, éste intenta utilizar la conexión especificada en la definición de canal. Para que tenga éxito, es necesario que la conexión esté definida y disponible. En esta sección se explica cómo hacerlo.

Al configurar la comunicación para IBM MQ en AIX and Linux, se puede elegir entre los siguientes tipos de comunicación:

- TCP/IP
- LU6.2

Cada definición de canal debe especificar sólo una como el atributo de protocolo de transmisión (Tipo de transporte). Un gestor de colas puede utilizar uno o más protocolos.

Para IBM MQ MQI clients, podría ser conveniente tener canales alternativos que utilicen protocolos de transmisión diferentes. Consulte IBM MQ MQI clients.

Procedimiento

Para obtener información sobre cómo configurar la comunicación para el sistema AIX o Linux, consulte el subtema del tipo de comunicación elegido:

- “Definición de una conexión TCP en AIX and Linux” en la página 282
- “Definición de una conexión LU 6.2 en AIX and Linux” en la página 286
- MQ Adv. MQ Adv. VUE “Definición de una conexión de Aspera gateway en plataformas Linux o Windows” en la página 887

Tareas relacionadas

“Supervisión y control de canales en AIX, Linux, and Windows” en la página 263

Para DQM debe crear, supervisar y controlar los canales con los gestores de colas remotos. Puede controlar los canales utilizando mandatos, programas, IBM MQ Explorer, archivos para las definiciones de canal y un área de almacenamiento para la información de sincronización.

"Configuración de conexiones entre el cliente y el servidor" en la página 16
Para configurar los enlaces de comunicación entre IBM MQ MQI clients y servidores, decida el protocolo de comunicación, defina las conexiones en ambos extremos del enlace, inicie un escucha y defina canales.

"Configuración de la comunicación en Windows" en la página 273
Cuando se inicia un canal de gestión de colas distribuidas, éste intenta utilizar la conexión especificada en la definición de canal. Para que esto tenga éxito, la conexión debe estar definida y disponible. En esta sección se explica cómo hacerlo empleando las formas de comunicación disponibles en sistemas IBM MQ for Windows.

Referencia relacionada

"Qué tipo de comunicación utilizar" en la página 17
Diferentes plataformas dan soporte a diferentes protocolos de comunicación. El protocolo de transmisión que elija dependerá de su combinación de plataformas de servidor y IBM MQ MQI client.

Linux AIX *Definición de una conexión TCP en AIX and Linux*

La definición de canal en el extremo emisor especifica la dirección del destino. El daemon de escucha o inet está configurado para la conexión en el extremo receptor.

Antes de empezar

MQ Adv. CD Un canal de mensajes que utiliza TCP/IP puede apuntar a un IBM Aspera faspio Gateway, que proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de la red. Un gestor de colas que se ejecuta en cualquier plataforma autorizada puede conectarse a través de un Aspera gateway. La propia pasarela se despliega en Red Hat o Ubuntu Linux, o Windows. Consulte Definición de una conexión de Aspera gateway en Linux o Windows.

Extremo emisor

Especifique el nombre de host, o la dirección TCP de la máquina de destino, en el campo Nombre de conexión de la definición de canal. El puerto de conexión toma el valor predeterminado 1414. El número de puerto 1414 es el asignado por la IANA (Internet Assigned Numbers Authority) para IBM MQ.

Para utilizar un número de puerto distinto del predeterminado, cambie el campo Nombre de conexión de este modo:

```
Connection Name REMHOST(1822)
```

donde REMHOST es el nombre de host de la máquina remota y 1822 es el número de puerto obligatorio. (Debe ser el puerto en el que el escucha del extremo receptor está a la escucha).

Opcionalmente, puede cambiar el número de puerto especificándolo en el archivo de configuración del gestor de colas (qm.ini):

```
TCP:
Port=1822
```

Si desea más información sobre los valores que puede definir utilizando qm.ini, consulte Stanzas del archivo de configuración para la gestión de colas distribuida.

Recepción en TCP

Puede utilizar el escucha de TCP/IP, que es el daemon inet (inetd), o el escucha de IBM MQ.

Algunas distribuciones de Linux utilizan el daemon inet ampliado (xinetd) en lugar del daemon inet. Para obtener más información sobre cómo utilizar el daemon inet ampliado en un sistema Linux , consulte el Paso 2 de Ejemplo: configuración de la comunicación entre plataformas IBM MQ en Linux.

Conceptos relacionados

"Utilización del escucha TCP/IP en AIX and Linux" en la página 283
Para iniciar los canales en AIX and Linux, deben editarse el archivo `/etc/services` y el archivo `inetd.conf`

"Utilización de la opción de proceso de escucha TCP en IBM MQ for Multiplatforms" en la página 284
En TCP, las conexiones se tratan de forma incompleta a menos que tenga lugar un reconocimiento entre el servidor y el cliente. Estas conexiones se llaman solicitudes de conexión pendientes. Se establece un valor máximo para estas solicitudes de conexión pendientes y se puede considerar una reserva de solicitudes en espera del puerto TCP para que el escucha acepte la solicitud.

"Utilización del escucha de IBM MQ" en la página 285
Para ejecutar el escucha proporcionado con IBM MQ, que inicia nuevos canales como hebras, utilice el mandato `runmqlsr`.

"Utilización de la opción TCP/IP SO_KEEPALIVE" en la página 285
En algunos sistemas AIX and Linux, puede definir cuánto tiempo espera TCP antes de comprobar si la conexión sigue estando disponible y con qué frecuencia intenta de nuevo la conexión si la primera comprobación falla. Esto puede ser un parámetro ajustable del kernel o se puede entrar en la línea de mandatos.

Linux AIX *Utilización del escucha TCP/IP en AIX and Linux*

Para iniciar los canales en AIX and Linux, deben editarse el archivo `/etc/services` y el archivo `inetd.conf`

Siga estas instrucciones:

1. Edite el archivo `/etc/services`:

   Nota: Para editar el archivo `/etc/services`, debe haber iniciado la sesión como superusuario o root. Puede cambiarlo, pero debe coincidir con el número de puerto especificado en el extremo emisor.

   Añada la línea siguiente al archivo:

   ```
   MQSeries 1414/tcp
   ```

   donde 1414 es el número de puerto que IBM MQ necesita. El número de puerto no debe exceder 65535.
2. Añada una línea en el archivo `inetd.conf` para llamar al programa amqcrsta, donde *MQ_INSTALLATION_PATH* representa el directorio de alto nivel en el que IBM MQ está instalado:

   ```
   MQSeries stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta
   [-m Queue_Man_Name]
   ```

Las actualizaciones están activas después de que inetd haya vuelto a leer los archivos de configuración. Para ello, emita los mandatos siguientes desde el ID de usuario root:

- AIX En AIX:

  ```
  refresh -s inetd
  ```

- Linux En los sistemas Linux:

  ```
  kill -1 process_number
  ```

Cuando el programa de escucha iniciado por inetd hereda el entorno local de inetd, es posible que MQMDE no se haya respetado (fusionado) y esté colocado en la cola como datos de mensaje. Para asegurarse de que el MQMDE se respeta, debe establecer el entorno local correctamente. Puede que el entorno local establecido por inetd no coincida con otros entornos locales utilizados por los procesos de IBM MQ. Para establecer el entorno local:

1. Cree un script de shell que establezca las variables de entorno de entorno local LANG, LC_COLLATE, LC_CTYPE, LC_MONETARY, LC_NUMERIC, LC_TIME y LC_MESSAGES en el entorno local utilizado para otro proceso de IBM MQ.
2. En el mismo script de shell, llame al programa de escucha.
3. Modifique el archivo `inetd.conf` para llamar al script de shell en lugar del programa de escucha.

Es posible tener más de un gestor de colas en el servidor. Debe añadir una línea a cada uno de los dos archivos, para cada uno de los gestores de colas. Por ejemplo:

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

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

donde *MQ_INSTALLATION_PATH* representa el directorio de alto nivel en el que está instalado IBM MQ.

Esto evita que se generen mensajes de error si hay una limitación en el número de solicitudes de conexión pendientes en cola en un único puerto TCP. Para obtener información sobre el número de solicitudes de conexión pendientes, consulte “Utilización de la opción de proceso de escucha TCP en IBM MQ for Multiplatforms” en la página 284.

Multi *Utilización de la opción de proceso de escucha TCP en IBM MQ for Multiplatforms*

En TCP, las conexiones se tratan de forma incompleta a menos que tenga lugar un reconocimiento entre el servidor y el cliente. Estas conexiones se llaman solicitudes de conexión pendientes. Se establece un valor máximo para estas solicitudes de conexión pendientes y se puede considerar una reserva de solicitudes en espera del puerto TCP para que el escucha acepte la solicitud.

Los valores predeterminados de reserva del escucha se muestran en la Tabla 24 en la página 284.

Tabla 24. Número máximo de solicitudes de conexión pendientes puestas en cola en un puerto TCP/IP

Plataforma de servidor	**Número máximo de solicitudes de conexión**
AIX AIX	100
Linux Linux	100
IBM i IBM i	255
Windows Windows Servidor	100

Si la reserva alcanza los valores mostrados en la Tabla 24 en la página 284, la conexión TCP/IP se rechaza y el canal no se puede iniciar.

Para canales MCA, esto da como resultado que el canal entra en un estado RETRY e intenta volver a conectarse en un momento posterior.

Sin embargo, para evitar este error, puede añadir una entrada en el archivo `qm.ini`:

```
TCP:
ListenerBacklog = n
```

Esto altera temporalmente el número máximo predeterminado de solicitudes pendientes (consulte Tabla 24 en la página 284) para el escucha TCP/IP.

Nota: Algunos sistemas operativos dan soporte a un valor mayor que el predeterminado. Si es necesario, este valor se puede utilizar para evitar alcanzar el límite de conexiones.

Para ejecutar el escucha con la opción `backlog` habilitada:

- Utilice el mandato `runmqlsr -b` o
- Utilice el mandato de MQSC **`DEFINE LISTENER`** con el atributo BACKLOG establecido en el valor necesario.

Para obtener más información sobre el mandato **`runmqlsr`**, consulte runmqlsr. Para obtener más información sobre el mandato DEFINE LISTENER, consulte DEFINE LISTENER.

Conceptos relacionados

"Using the TCP listener backlog option on z/OS" en la página 998
When receiving on TCP/IP, a maximum number of outstanding connection requests is set. These outstanding requests can be considered a *backlog* of requests waiting on the TCP/IP port for the listener to accept the request.

Linux AIX *Utilización del escucha de IBM MQ*

Para ejecutar el escucha proporcionado con IBM MQ, que inicia nuevos canales como hebras, utilice el mandato `runmqlsr`.

Por ejemplo:

```
runmqlsr -t tcp [-m QMNAME] [-p 1822]
```

Los corchetes indican parámetros opcionales; QMNAME no es necesario para el gestor de colas predeterminado y el número de puerto no es necesario si está utilizando el valor predeterminado (1414). El número de puerto no debe exceder 65535.

Para obtener un rendimiento óptimo, ejecute el escucha de IBM MQ como una aplicación de confianza, como se describe en "Ejecución de canales y escuchas como aplicaciones de confianza" en la página 262. Consulte Restricciones para aplicaciones de confianza para obtener información sobre aplicaciones de confianza.

Puede detener todos los escuchas de IBM MQ que se ejecutan en un gestor de colas que está inactivo mediante el mandato:

```
endmqlsr [-m QMNAME]
```

Si no especifica un nombre de gestor de colas, se toma el gestor de colas predeterminado.

Linux AIX *Utilización de la opción TCP/IP SO_KEEPALIVE*

En algunos sistemas AIX and Linux, puede definir cuánto tiempo espera TCP antes de comprobar si la conexión sigue estando disponible y con qué frecuencia intenta de nuevo la conexión si la primera comprobación falla. Esto puede ser un parámetro ajustable del kernel o se puede entrar en la línea de mandatos.

Si desea utilizar la opción SO_KEEPALIVE (si desea más información, consulte "Cómo comprobar que el otro extremo del canal sigue estando disponible" en la página 247) debe añadir la entrada siguiente al archivo de configuración del gestor de colas (qm.ini):

```
TCP:
KeepAlive=yes
```

Consulte la documentación del sistema AIX o Linux para obtener más información.

Linux AIX *Definición de una conexión LU 6.2 en AIX and Linux*

SNA debe configurarse de manera que pueda establecerse una conversación LU 6.2 entre las dos máquinas.

Para obtener la información más reciente sobre la configuración de SNA sobre TCP/IP, consulte la siguiente documentación en línea de IBM: Communications Server.

SNA debe configurarse de modo que pueda establecerse una conversación LU 6.2 entre los dos sistemas.

Para obtener más información, consulte la publicación *Multiplatform APPC Configuration Guide* y la tabla siguiente.

Tabla 25. Valores del sistema AIX o Linux local para una plataforma de gestor de colas remoto

Plataforma remota	**TPNAME**	**TPPATH**
z/OS sin CICS	El mismo que el TPName correspondiente en la información complementaria sobre el gestor de colas remoto.	-
z/OS utilizando CICS	CRCK (emisor) CKSV (peticionario) CRCK (servidor)	-
IBM i	El mismo que el valor de comparación de la entrada de direccionamiento del sistema IBM i.	-
Sistemas AIX and Linux	El mismo que el TPName correspondiente en la información complementaria sobre el gestor de colas remoto.	*MQ_INSTALLATION_PATH*/bin/amqcrs6a
Windows	El mismo que el especificado en el mandato Run Listener de Windows, o el programa de transacción invocable definido mediante TpSetup en Windows.	*MQ_INSTALLATION_PATH*\bin\amqcrs6a

MQ_INSTALLATION_PATH representa el directorio de alto nivel en el que está instalado IBM MQ.

Si tiene más de un gestor de colas en la misma máquina, asegúrese de que los TPnames de las definiciones de canal son exclusivos.

Conceptos relacionados

"Extremo emisor en LU 6.2 en AIX and Linux" en la página 286
En sistemas AIX and Linux, cree un objeto complementario CPI-C (destino simbólico) y especifique este nombre en el campo Nombre de conexión de la definición de canal. Cree también un enlace LU 6.2 al socio.

"Recepción en LU 6.2 en AIX and Linux" en la página 287
En sistemas AIX and Linux, cree una conexión de escucha en el extremo receptor y un perfil de conexión lógica LU 6.2, así como un perfil TPN.

Linux AIX *Extremo emisor en LU 6.2 en AIX and Linux*

En sistemas AIX and Linux, cree un objeto complementario CPI-C (destino simbólico) y especifique este nombre en el campo Nombre de conexión de la definición de canal. Cree también un enlace LU 6.2 al socio.

En el objeto complementario CPI-C, especifique el nombre de LU asociada en la máquina receptora, el nombre de programa de transacción el nombre de modalidad. Por ejemplo:

```
Partner LU Name              REMHOST
Remote TP Name               recv
Service Transaction Program  no
Mode Name                    #INTER
```

Se utiliza SECURITY PROGRAM, siempre que esté soportado por CPI-C, cuando IBM MQ intenta establecer una sesión SNA.

Linux AIX *Recepción en LU 6.2 en AIX and Linux*

En sistemas AIX and Linux, cree una conexión de escucha en el extremo receptor y un perfil de conexión lógica LU 6.2, así como un perfil TPN.

En el perfil TPN, entre la vía de acceso completa al archivo ejecutable y el nombre de programa de transacción:

```
Full path to TPN executable    MQ_INSTALLATION_PATH/bin/amqcrs6a
Transaction Program name       recv
User ID                        0
```

MQ_INSTALLATION_PATH representa el directorio de alto nivel en el que está instalado IBM MQ.

En sistemas en los que puede establecer el ID de usuario, especifique un usuario que sea miembro del grupo mqm.

AIX AIX, establezca las variables de entorno APPCTPN (nombre de transacción) y APPCLLU (nombre de LU local) (puede utilizar los paneles de configuración para el programa de transacción invocado).

Es posible que necesite utilizar un gestor de colas distinto del gestor de colas predeterminado. Si es así, defina un archivo de mandatos que llame a:

```
amqcrs6a -m Queue_Man_Name
```

A continuación, llame al archivo de mandatos.

IBM i Supervisión y control de canales en IBM i

Utilice los mandatos y paneles de DQM para crear, supervisar y controlar los canales con gestores de colas remotos. Cada gestor de colas tiene un programa DQM para controlar las interconexiones con gestores de colas remotos compatibles.

Acerca de esta tarea

La lista siguiente consiste en una breve descripción de los componentes de la función de control de canales:

- Las definiciones de canal se almacenan como objetos del gestor de colas.
- Los mandatos de canal son un subconjunto del conjunto de mandatos de IBM MQ for IBM i.

  Utilice el mandato GO CMDMQM para visualizar el conjunto completo de mandatos de IBM MQ for IBM i.
- Puede utilizar los paneles o los mandatos de definición de canal para:
  - Crear, copiar, visualizar, cambiar y suprimir definiciones de canal
  - Iniciar y detener canales, ejecutar mandatos ping, restablecer números de secuencia del canal y resolver los mensajes pendientes cuando no es posible restablecer los enlaces
  - Mostrar información de estado sobre los canales
- Los canales también se pueden gestionar utilizando MQSC
- Los canales también se pueden gestionar utilizando IBM MQ Explorer

- Los números de secuencia y los identificadores de *unidad lógica de trabajo (LUW)* se almacenan en el archivo de sincronización y se utilizan para fines de sincronización de canal.

Puede utilizar los mandatos y los paneles para: definir los canales de mensajes y los objetos asociados, y supervisar y controlar los canales de mensajes. Mediante la tecla F4=Solicitud se puede especificar el gestor de colas pertinente. Si no utiliza esta tecla, se utiliza el gestor de colas predeterminado. Con F4=Solicitud, se muestra un panel adicional en el que puede especificarse el nombre del gestor de colas pertinente y, a veces, otros datos.

Los objetos que necesita definir con los paneles son:

- Colas de transmisión
- Definiciones de colas remotas
- Definiciones de alias de gestor de colas
- Definiciones de alias de colas de respuesta
- Colas locales de respuestas
- Definiciones de canal de mensajes

Para obtener más información sobre los conceptos implicados en el uso de estos objetos, consulte "Configuración de la gestión de colas distribuidas" en la página 210.

Los canales deben estar completamente definidos y sus objetos asociados deben existir y poder utilizarse antes de poder iniciar un canal.

Además, el enlace de comunicación concreto de cada canal debe estar definido y disponible antes de poder ejecutar el canal. Para obtener una descripción de cómo se definen los enlaces LU 6.2 y TCP/IP, consulte la guía de comunicaciones particular para su instalación.

Procedimiento

- Para obtener más información acerca de cómo crear y trabajar con objetos, consulte:
  - "Creación de objetos en IBM i" en la página 289
  - "Creación de un canal en IBM i" en la página 289
  - "Inicio de un canal en IBM i" en la página 291
  - "Selección de un canal en IBM i" en la página 291
  - "Examinar un canal en IBM i" en la página 292
  - "Renombrar un canal en IBM i" en la página 294
  - "Trabajar con estado de canal en IBM i" en la página 294
  - "Opciones de Trabajar con canales en IBM i" en la página 295

Conceptos relacionados

"Configuración de la comunicación para IBM i" en la página 301
Cuando se inicia un canal de gestión de colas distribuidas, éste intenta utilizar la conexión especificada en la definición de canal. Para que tenga éxito, es necesario que la conexión esté definida y disponible.

Tareas relacionadas

"Configuración de conexiones entre el cliente y el servidor" en la página 16
Para configurar los enlaces de comunicación entre IBM MQ MQI clients y servidores, decida el protocolo de comunicación, defina las conexiones en ambos extremos del enlace, inicie un escucha y defina canales.

Referencia relacionada

Configuración de ejemplo - IBM MQ for IBM i

Ejemplo de planificación de canal de mensajes para IBM MQ for IBM i

Mandatos CL de IBM MQ for IBM i

IBM i Creación de objetos en IBM i

Puede utilizar el mandato CRTMQMQ para crear los objetos de colas y alias.

Puede crear los objetos de colas y alias, tales como; colas de transmisión, definiciones de colas remotas, definiciones de alias del gestor de colas, definiciones de alias de respuesta y colas locales de respuesta.

Para obtener una lista de objetos predeterminados, consulte Objetos predeterminados y del sistema.

IBM i Creación de un canal en IBM i

Puede crear un canal desde el panel Crear Canal o mediante el mandato CRTMQMCHL en la línea de mandatos.

Para crear un canal:

1. Utilice F6 desde el panel Trabajar con canales MQM (WRKMQMCHL).

   O bien, utilice el mandato CRTMQMCHL de la línea de mandatos.

   De cualquier modo, aparece el panel Crear canal. Tipo:

   - El nombre del canal en el campo proporcionado
   - El tipo de canal para este extremo del enlace
2. Pulse Intro.

Nota: Debe nombrar todos los canales en la red de forma exclusiva. Como se muestra en Diagrama de red que muestra todos los canales, incluir los nombres de los gestores de colas de origen y destino en el nombre del canal es una buena forma de hacerlo.

Las entradas se validan y se informa de errores de forma inmediata. Corrija los errores y continúe.

Aparece el panel de valores de canal adecuados para el tipo de canal que ha elegido. Complete los campos con la información que ha recopilado anteriormente. Pulse Intro para crear el canal.

En los paneles de ayuda, y en Atributos de canal, se le ofrece ayuda para decidir el contenido de diversos campos en las descripciones de los paneles de definiciones de canal.

```
Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

Channel name . . . . . > CHANNAME_____________
Channel type . . . . . > *SDR__    *RCVR, *SDR, *SVR, *RQSTR...
Message Queue Manager name     *DFT_____________________________
_____
Replace . . . . . .  *NO_     *NO, *YES
Transport type . . . . . .  *TCP____   *LU62, *TCP, *SYSDFTCHL
Text 'description' . . . . . > 'Example Channel Definition'_____________
_____________________________
Connection name . . . . . .  *SYSDFTCHL_____________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
_____________________________________________
More...
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
```

Figura 25. Crear canal (1)

```
Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

Transmission queue . . . . . .  'TRANSMISSION_QUEUE_NAME'_____________________
_____________________
Message channel agent . . . . .  *NONE_______  Name, *SYSDFTCHL, *NONE
Library . . . . . .  __________  Name
Message channel agent user ID .  *SYSDFTCHL__ Character value...
Coded Character Set Identifier   *SYSDFTCHL__ 0-9999, *SYSDFTCHL
Batch size . . . . . .  50_________  1-9999, *SYSDFTCHL
Disconnect interval . . . . . .  6000_______  1-999999, *SYSDFTCHL
Short retry interval . . . . . .  60_________  0-999999999, *SYSDFTCHL
Short retry count . . . . . .  10_________  0-999999999, *SYSDFTCHL
Long retry interval . . . . . .  1200_______  0-999999999, *SYSDFTCHL
Long retry count . . . . . .  999999999__  0-999999999, *SYSDFTCHL
Security exit . . . . . .  *NONE_____  Name, *SYSDFTCHL, *NONE
Library . . . . . .  __________  Name
Security exit user data . . . .  *SYSDFTCHL_____________________

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

Figura 26. Crear canal (2)

```
Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

Send exit . . . . . .  *NONE_______  Name, *SYSDFTCHL, *NONE
Library . . . . . .  ___________ Name
+ for more values   __________
Send exit user data . . . . . .  _____________________________
+ for more values  _____________________________
Receive exit . . . . . .  *NONE_____  Name, *SYSDFTCHL, *NONE
Library . . . . . .  __________  Name
+ for more values  __________
__________
Receive exit user data . . . . .  _____________________________
+ for more values  _____________________________
Message exit . . . . . .  *NONE_____  Name, *SYSDFTCHL, *NONE
Library . . . . . .  __________  Name
+ for more values  __________

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

Figura 27. Crear canal (3)

```
Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

Message exit user data . . . . .  _________________________________
+ for more values  _________________________________
Convert message . . . . . .  *SYSDFTCHL_  *YES, *NO, *SYSDFTCHL
Sequence number wrap . . . . . .  999999999__  100-999999999, *SYSDFTCHL
Maximum message length . . . . .  4194304____  0-4194304, *SYSDFTCHL
Heartbeat interval . . . . . .  300________  0-999999999, *SYSDFTCHL
Non Persistent Message Speed . .  *FAST_____  *FAST, *NORMAL, *SYSDFTCHL
Password . . . . . .  *SYSDFTCHL_  Character value, *BLANK...
Task User Profile . . . . . .  *SYSDFTCHL_  Character value, *BLANK...
Transaction Program Name . . . .  *SYSDFTCHL




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

Figura 28. Crear canal (4)

IBM i Inicio de un canal en IBM i

Puede iniciar un canal desde el panel Trabajar con canales o mediante el mandato STRMQMCHL en la línea de mandatos.

Los escuchas sólo son válidos para TCP. Para escuchas SNA, debe configurar el subsistema de comunicaciones.

Para que las aplicaciones puedan intercambiar mensajes, debe iniciar un programa de escucha para conexiones de entrada mediante el mandato STRMQMLSR.

Para conexiones de salida, debe iniciar el canal de una de las maneras siguientes:

1. Utilice el mandato de CL STRMQMCHL, especificando un nombre de canal, para iniciar el canal como proceso o hebra, en función del parámetro MCATYPE. (Si los canales se inician como hebras, son hebras de un iniciador de canal.)

   ```
   STRMQMCHL CHLNAME(QM1.TO.QM2) MQNAME(MYQMGR)
   ```

2. Utilice un iniciador de canal para desencadenar el canal. Un iniciador de canal se inicia automáticamente cuando se inicia el gestor de colas. Este inicio automático se puede eliminar cambiando la stanza chinit en el archivo qm.ini para dicho gestor de colas.
3. Utilice el mandato WRKMQMCHL para iniciar el panel Trabajar con canales y elija la opción 14 para iniciar un canal.

IBM i Selección de un canal en IBM i

Puede seleccionar un canal desde el panel Trabajar con canales.

Para seleccionar un canal, utilice el mandato WRKMQMCHL para empezar en el panel Trabajar con Canales:

1. Mueva el curso al campo de opción asociado con el nombre de canal necesario.
2. Escriba un número de opción.
3. Presione Intro para activar su elección.

Si selecciona más de un canal, las opciones se activan en secuencia.

```
Work with MQM Channels

Queue Manager Name . . :  CNX

Type options, press Enter.
2=Change  3=Copy  4=Delete  5=Display  8=Work with Status  13=Ping
14=Start  15=End  16=Reset  17=Resolve

Opt   Name           Type    Transport   Status
CHLNIC          *RCVR    *TCP     INACTIVE
CORSAIR.TO.MUSTANG    *SDR    *LU62    INACTIVE
FV.CHANNEL.MC.DJE1    *RCVR    *TCP     INACTIVE
FV.CHANNEL.MC.DJE2    *SDR    *TCP     INACTIVE
FV.CHANNEL.MC.DJE3    *RQSTR   *TCP     INACTIVE
FV.CHANNEL.MC.DJE4    *SVR    *TCP     INACTIVE
FV.CHANNEL.PETER     *RCVR    *TCP     INACTIVE
FV.CHANNEL.PETER.LU   *RCVR    *LU62    INACTIVE
FV.CHANNEL.PETER.LU1   *RCVR    *LU62    INACTIVE
More...
Parameters or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F6=Create  F9=Retrieve  F12=Cancel
F21=Print
```

Figura 29. Trabajar con canales

IBM i Examinar un canal en IBM i

Puede examinar un canal desde el panel Visualizar canal o mediante el mandato DSPMQMCHL en la línea de mandatos.

Para examinar la configuración de un canal, utilice el mandato WRKMQMCHL para empezar en el panel Visualizar canal:

1. Opción de tipo 5 (Visualizar) en el nombre de canal necesario.
2. Presione Intro para activar su elección.

Si selecciona más de un canal, se presentan en secuencia.

O bien, puede utilizar el mandato DSPMQMCHL desde la línea de mandatos.

Esto hace que se visualice el panel Visualizar canal adecuado con detalles de los valores actuales para el canal. Los campos se describen en la sección Atributos de canal.

```
Display MQM Channel

Channel name . . . . . :  ST.JST.2TO1
Queue Manager Name . . . . . :  QMREL
Channel type . . . . . :  *SDR
Transport type . . . . . :  *TCP
Text 'description' . . . . . :  John's sender to WINSDOA1

Connection name . . . . . :  MUSTANG


Transmission queue . . . . . :  WINSDOA1

Message channel agent . . . . :
Library . . . . . :
Message channel agent user ID :  *NONE
Batch interval . . . . . :  0
Batch size . . . . . :  50
Disconnect interval . . . . . :  6000



F3=Exit  F12=Cancel  F21=Print
```

Figura 30. Visualizar un canal TCP/IP (1)

```
Display MQM Channel

Short retry interval . . . . . :  60
Short retry count . . . . . :  10
Long retry interval . . . . . :  6000
Long retry count . . . . . :  10
Security exit . . . . . :
Library . . . . . :
Security exit user data . . . :
Send exit . . . . . :
Library . . . . . :
Send exit user data . . . . . :
Receive exit . . . . . :
Library . . . . . :
Receive exit user data . . . . :
Message exit . . . . . :
Library . . . . . :
Message exit user data . . . . :
More...



F3=Exit  F12=Cancel  F21=Print
```

Figura 31. Visualizar un canal TCP/IP (2)

```
Display MQM Channel

Sequence number wrap . . . . . :  999999999
Maximum message length . . . . :  10000
Convert message . . . . . :  *NO
Heartbeat interval . . . . . .  300
Nonpersistent message speed . .  *FAST









Bottom



F3=Exit  F12=Cancel  F21=Print
```

Figura 32. Visualizar un canal TCP/IP (3)

IBM i Renombrar un canal en IBM i

Puede renombrar un canal desde el panel Trabajar con canales.

Para renombrar un canal de mensajes, empiece en el panel Trabajar con canales:

1. Finalice el canal.
2. Utilice la opción 3 (Copiar) para crear un duplicado con el nuevo nombre.
3. Utilice la opción 5 (Visualizar) para comprobar si se ha creado correctamente.
4. Utilice la opción 4 (Suprimir) para suprimir el canal original.

Si decide renombrar un canal de mensajes, asegúrese de que ambos extremos del canal estén renombrados simultáneamente.

IBM i Trabajar con estado de canal en IBM i

Puede trabajar con el estado del canal desde el panel Trabajar con estado de canal.

Utilice el mandato WRKMQMCHST para visualizar el primero de un conjunto de paneles que muestran el estado de los canales. Puede ver los paneles de estado en secuencia cuando selecciona Cambiar vista (F11).

O bien, si selecciona la opción 8 (Trabajar con estado) desde el panel Trabajar con canales MQM también se visualiza el primer panel de estado.

```
MQSeries Work with Channel Status

Type options, press Enter.
5=Display  13=Ping  14=Start  15=End  16=Reset  17=Resolve


Opt Name          Connection      Indoubt   Last Seq
CARTS_CORSAIR_CHAN  GBIBMIYA.WINSDOA1    NO        1
CHLNIC        9.20.2.213       NO        3
FV.CHANNEL.PETER2   9.20.2.213       NO      6225
JST.1.2        9.20.2.201       NO       28
MP_MUST_TO_CORS    9.20.2.213       NO       100
MUSTANG.TO.CORSAIR  GBIBMIYA.WINSDOA1    NO       10
MP_CORS_TO_MUST    9.20.2.213       NO       101
JST.2.3        9.5.7.126        NO       32
PF_WINSDOA1_LU62   GBIBMIYA.IYA80020    NO       54
PF_WINSDOA1_LU62   GBIBMIYA.WINSDOA1    NO       500
ST.JCW.EXIT.2TO1.CHL 9.20.2.213       NO       216

Bottom
Parameters or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F6=Create  F9=Retrieve  F11=Change view
F12=Cancel  F21=Print
```

Figura 33. Primero del conjunto de paneles de estado de canal

Las opciones disponibles en el panel Trabajar con estado de canal son:

Opción de menú	Descripción
5=Display	Visualiza la configuración del canal.
13=Ping	Inicia una acción de sondeo cuando proceda.
14=Start	Inicia el canal.
15=End	Detiene el canal.
16=Reset	Restablece el número de secuencia del canal.
17=Resolve	Resuelve en una situación de canal pendiente, de forma manual.

IBM i Opciones de Trabajar con canales en IBM i

El panel Trabajar con canales se alcanza mediante el mandato WRKMQMCHL y le permite supervisar el estado de todos los canales listados y emitir de nuevo los mandatos en los canales seleccionados.

Las opciones disponibles en el panel Trabajar con canales son:

Opción de menú	Descripción
“2=Change” en la página 296	Cambia los atributos de un canal.
“3=Copy” en la página 296	Copia los atributos de un canal a un canal nuevo.
“4=Delete” en la página 296	Suprime un canal.
“5=Display” en la página 296	Visualiza los valores actuales del canal.
“6=Create” en la página 297	Visualiza el panel Crear canal.
“8=Trabajar con estado” en la página 297	Visualiza los paneles de estado de canal.

Opción de menú	Descripción
“13=Ping” en la página 298	Ejecuta el recurso Sondear para probar la conexión con el sistema adyacente intercambiando un mensaje de datos fijo con el extremo remoto.
“14=Start” en la página 298	Inicia el canal seleccionado o restablece un canal receptor inhabilitado.
“15=End” en la página 299	Solicita al canal que se cierre.
“16=Reset” en la página 300	Solicita al canal restablecer los números de secuencia en este extremo del enlace. Los números deben ser iguales en ambos extremos para que el canal se inicie.
“17=Resolve” en la página 300	Solicita al canal resolver mensajes pendientes sin establecer la conexión con el otro extremo.
“18=Visualizar autorización” en la página 300	Muestra la autorización sobre objetos de IBM MQ
“19=Otorgar autorización” en la página 300	Otorga autorización sobre objetos de IBM MQ
“20=Revocar autorización” en la página 301	Revoca la autorización sobre objetos de IBM MQ
“21=Recuperar objeto” en la página 301	Recupera el objeto de IBM MQ
“22=Registrar imagen” en la página 301	Registra la imagen del objeto de IBM MQ

IBM i 2=Change

Utilice la opción Cambiar para cambiar una definición de canal existente.

La opción Cambiar o el mandato CHGMQMCHL cambia una definición de canal existente, excepto para el nombre de canal. Escriba encima de los campos que se van a cambiar en el panel de definición de canal y, a continuación, guarde la definición actualizada pulsando Intro.

IBM i 3=Copy

Utilice la opción Copiar para copiar un canal existente.

La opción Copiar utiliza el mandato CPYMQMCHL para copiar un canal existente. El panel Copiar le permite definir el nuevo nombre del canal. Sin embargo, debe restringir los caracteres utilizados a aquellos que sean válidos para nombres de objeto de IBM i; consulte Administración de IBM MQ for IBM i.

Pulse Intro en el panel Copiar para visualizar los detalles de los valores actuales. Puede cambiar cualquiera de los nuevos valores de canal. Guarde la definición de canal nuevo pulsando Intro.

IBM i 4=Delete

Utilice la opción Suprimir para suprimir el canal seleccionado.

Se visualiza un panel para confirmar o cancelar la solicitud.

IBM i 5=Display

Utilice la opción Visualizar para visualizar las definiciones actuales para el canal.

Esta opción visualiza el panel con los campos que muestran los valores actuales de los parámetros y está protegida para evitar la entrada de datos del usuario.

IBM i 6=Create

Utilice la opción Crear para visualizar el panel Crear canal.

Utilice la opción Crear o entre el mandato CRTMQMCHL desde la línea de mandatos, para obtener el panel Crear canal. Hay ejemplos de paneles Crear canal, que empiezan en la Figura 25 en la página 289.

Con este panel, puede crear una definición de canal desde una pantalla de campos rellenados con valores predeterminados suministrados por IBM MQ for IBM i. Escriba el nombre del canal, seleccione el tipo de canal que está creando y el método de comunicación que se utilizará.

Cuando pulse Intro, se visualizará el panel. Escriba información en todos los campos necesarios en este panel y los paneles restantes y, a continuación, guarde la definición pulsando Intro.

El nombre del canal debe ser el mismo en ambos extremos del canal y exclusivo dentro de la red. Sin embargo, debe restringir los caracteres utilizados a aquellos caracteres que sean válidos para nombres de objeto de IBM MQ for IBM i.

Todos los paneles tienen valores predeterminados proporcionados por IBM MQ for IBM i para algunos campos. Puede personalizar estos valores o bien cambiarlos cuando cree o copie canales. Para personalizar los valores, consulte la publicación *IBM MQ for IBM i System Administration*.

Puede crear su propio conjunto de valores predeterminados de canal configurando canales ficticios con los valores predeterminados necesarios para cada tipo de canal y copiándolos cada vez que desee crear nuevas definiciones de canal.

Referencia relacionada

Atributos de canal

IBM i 8=Trabajar con estado

Utilice Trabajar con estado para ver información detallada sobre el estado de canal.

La columna estado indica si el canal está activo o inactivo y se visualiza continuamente en el panel Trabajar con canales MQM. Utilice la opción 8 (Trabajar con Estado) para ver más información de estado visualizada. De forma alternativa, esta información se puede visualizar desde la línea de mandatos con el mandato WRKMQMCHST. Consulte “Trabajar con estado de canal en IBM i” en la página 294.

- Nombre de canal
- Tipo de canal
- Estado de canal
- Instancia de canal
- Gestor de colas remoto
- Nombre de cola de transmisión
- Nombre de la conexión de comunicaciones
- Estado pendiente del canal
- Último número de secuencia
- Número de mensajes pendientes
- Número de secuencia pendiente
- Número de mensajes en la cola de transmisión
- Identificador de unidad lógica de trabajo
- Identificador de unidad lógica de trabajo pendiente
- Canal Substate
- Supervisión de canal
- Compresión de cabecera
- Compresión de mensaje
- Indicador de hora de compresión

- Indicador de velocidad de compresión
- Indicador de hora de cola de transmisión
- Indicador de hora de red
- Indicador de hora de salida
- Indicador de tamaño de lote
- Conversaciones compartidas actuales
- Conversaciones máximas compartidas

IBM i *13=Ping*

Utilice la opción Ping para intercambiar un mensaje de datos fijo con el extremo remoto.

Un Ping de IBM MQ satisfactorio da cierta confianza al supervisor del sistema de que el canal está disponible y en funcionamiento.

Ping no implica el uso de colas de transmisión y colas de destino. Utiliza definiciones de canal, el enlace de comunicaciones relacionado y la configuración de la red.

Está disponible únicamente desde canales emisores y servidores. El canal correspondiente se inicia en el extremo del enlace y realiza la negociación del parámetro de inicio. Los errores se notifican con normalidad.

El resultado del intercambio de mensajes aparece en el panel Ping y se trata del texto del mensaje devuelto, junto con la hora a la que se envió el mensaje y la hora a la que se recibió la respuesta.

Sondeo con LU 6.2

Cuando se invoca Ping en IBM MQ for IBM i, se ejecuta con el ID de usuario del usuario que solicita la función, mientras que la manera habitual de ejecutar un programa de canal es que se utilice el ID de usuario QMQM para los programas de canal. El ID de usuario fluye a la parte receptora y debe ser válido en el extremo receptor para que se asigne la conversación LU 6.2.

IBM i *14=Start*

Utilice la opción Iniciar para iniciar un canal manualmente.

La opción Iniciar está disponible para canales emisores, servidores y peticionarios. No es necesario en el caso de que se haya configurado un canal con el desencadenamiento del gestor de colas.

La opción Iniciar también se utiliza para los canales receptor, de conexión con el servidor, emisor de clúster y receptor de clúster. Iniciar un canal receptor que está en estado STOPPED significa que se puede iniciar desde el canal remoto.

Cuando se inicia, el MCA emisor lee el archivo de definición de canal y abre la cola de transmisión. Se emite una secuencia de inicio de canal, que inicia de forma remota el MCA correspondiente del canal receptor o servidor. Cuando se han iniciado, los procesos del emisor y servidor esperan hasta que llegan mensajes a la cola de transmisión y los transmiten cuando lleguen.

Cuando se utiliza el desencadenamiento, debe iniciar el proceso desencadenante en ejecución continuamente para supervisar la cola de inicio. El mandato STRMQMCHLI se puede utilizar para iniciar el proceso.

En el extremo de un canal, el proceso receptor podría iniciarse en respuesta a un inicio de canal desde el extremo emisor. El método para realizarlo es diferente para canales conectados a LU 6.2 y TCP/IP:

- Los canales conectados a LU 6.2 no requieren ninguna acción explícita en el extremo receptor de un canal.
- Los canales conectados a TCP requieren que se ejecute un proceso de escucha continuo. Este proceso espera las solicitudes de inicio de canal del extremo remoto del enlace e inicia el proceso definido en las definiciones de canal para dicha conexión.

Cuando el sistema remoto es IBM i, puede utilizar el mandato STRMQMLSR.

El uso de la opción Iniciar hace que el canal se resincronice cuando sea necesario.

Para que el inicio se realice correctamente:

- Deben existir definiciones de canal, locales y remotas. Si no hay ninguna definición de canal adecuada para un canal receptor o de conexión con el servidor, se crea una predeterminada automáticamente si el canal está definido automáticamente. Consulte Programa de salida de definición automática de canal.
- La cola de transmisión debe existir, estar habilitada para GET y no tener ningún otro canal que la utilice.
- Los MCA, locales y remotos, deben existir.
- El enlace de comunicaciones debe estar disponible.
- Los gestores de colas deben estar en ejecución, locales y remotos.
- El canal de mensajes debe estar inactivo.

Para transferir mensajes, deben existir colas remotas y definiciones de colas remotas.

Se devuelve un mensaje al panel que confirma que la solicitud para iniciar un canal se ha aceptado. Para confirmar que el mandato de inicio se ha ejecutado correctamente, compruebe el registro del sistema o pulse F5 (renovar la pantalla).

IBM i ***15=End***

Utilice Finalizar para detener la actividad de canal

Utilice la opción Finalizar para solicitar al canal que detenga la actividad. El canal no envía ningún mensaje más.

Seleccione F4 antes de pulsar Intro para elegir si el canal está STOPPED o INACTIVE y si desea detener el canal utilizando una parada CONTROLLED o IMMEDIATE. Un canal detenido debe reiniciarse mediante el operador para volver a estar activo. Un canal inactivo se puede desencadenar.

Detención inmediata

Utilice la Detención inmediata para detener un canal sin realizar ninguna unidad de trabajo.

Esta opción termina el proceso del canal. Como resultado de ello, el canal no completa el proceso del lote actual de mensajes y no puede, por lo tanto, dejar el canal pendiente. En general, es mejor para los operadores utilizar la opción de detención controlada.

Detención controlada

Utilice Detención controlada para detener un canal al final de la unidad actual de trabajo.

Esta opción solicita al canal cerrarse de forma ordenada; el lote de mensajes actual se completa y el procedimiento del punto de sincronización se realiza con el otro extremo del canal.

Reinicio de canales detenidos

Cuando un canal pasa al estado STOPPED, debe reiniciar el canal manualmente. Puede reiniciar el canal de las formas siguientes:

- Utilizando el mandato MQSC **START CHANNEL**.
- Utilizando el mandato PCF **Start Channel** .
- Utilizando IBM MQ Explorer.
- z/OS En z/OS, utilizando el panel Iniciar un canal.
- IBM i En IBM i, utilizando el mandato **STRMQMCHL CL** o la opción **START** en el panel WRKMQMCHL.

Para canales emisores o servidores, cuando el canal ha entrado en el estado STOPPED, la cola de transmisión asociada se ha establecido en GET(DISABLED) y se ha desactivado el desencadenamiento. Cuando se recibe la solicitud de inicio, estos atributos se restablecen automáticamente.

z/OS Si el iniciador de canal se detiene mientras un canal está en estado RETRYING o STOPPED, el estado de canal se recuerda cuando se reinicia el iniciador de canal. Sin embargo, el estado del canal para el tipo de canal SVRCONN se restablece si el iniciador de canal se detiene mientras el canal está en estado STOPPED.

Multi Si el gestor de colas se detiene mientras un canal está en estado RETRYING o STOPPED, el estado de canal se recuerda cuando se reinicia el gestor de colas. A partir de IBM MQ 8.0, estos se aplica también a los canales SVRCONN. Anteriormente, el estado de canal para el tipo de canal SVRCONN se ha restablecido si el iniciador de canal se ha detenido mientras el canal estaba en estado STOPPED.

IBM i *16=Reset*

Utilice la opción Restablecer para forzar una nueva secuencia de mensaje.

La opción Restablecer cambia el número de secuencia de mensaje. Utilícela con cuidado y sólo después de haber utilizado la opción Resolver para resolver cualquier situación pendiente. Esta opción sólo está disponible en el canal emisor o servidor. El primer mensaje inicia la nueva secuencia la próxima vez que se inicia el canal.

IBM i *17=Resolve*

Utilice la opción Resolver para forzar una confirmación o restitución local de mensajes pendientes mantenidos en una cola de transmisión.

Utilice la opción Resolver cuando un emisor o servidor mantiene los mensajes pendientes; por ejemplo porque un extremo del enlace se ha terminado y no hay ninguna perspectiva de recuperarlo. La opción Resolver acepta uno de los parámetros: BACKOUT o COMMIT. La restitución restaura mensajes a la cola de transmisión, mientras que la confirmación los descarta.

El programa del canal no intenta establecer una sesión con un socio. En su lugar, determina el identificador de unidad lógica de trabajo (LUWID) que representa los mensajes pendientes. A continuación, emite, tal como se solicitó:

- BACKOUT para restaurar los mensajes a la cola de transmisión; o
- COMMIT para suprimir los mensajes de la cola de transmisión.

Para que la resolución se realice correctamente:

- El canal debe estar inactivo
- El canal debe estar pendiente
- El tipo de canal debe ser emisor o servidor
- La definición de canal, local, debe existir
- El gestor de colas debe estar en ejecución, local

IBM i *18=Visualizar autorización*

Utilice la opción Visualizar autorización para visualizar las acciones que un usuario está autorizado a realizar en un objeto de IBM MQ específico.

Para un objeto y un usuario seleccionados, el mandato DSPMQAUT muestra las autorizaciones que tiene el usuario para realizar acciones en un objeto de IBM MQ. Si el usuario es miembro de varios grupos, el mandato muestra las autorizaciones combinadas de todos los grupos para el objeto.

IBM i *19=Otorgar autorización*

Utilice la opción Otorgar autorización para otorgar la autorización para realizar acciones en objetos de IBM MQ a otro usuario o grupo de usuarios.

El mandato GRTMQMAUT sólo está disponible para usuarios del grupo QMQMADM. Un usuario en QMQMADM otorga autorización a otros usuarios para realizar acciones en los objetos de IBM MQ indicados en el mandato, identificando a los usuarios por su nombre u otorgando autorización a todos los usuarios en *PUBLIC.

IBM i *20=Revocar autorización*

Utilice Revocar autorización para eliminar la autorización para realizar acciones en objetos de los usuarios.

El mandato RVKMQMAUT sólo está disponible para los usuarios del grupo RVKMQMAUT. Un usuario del grupo QMQMADM elimina la autorización de otros usuarios para realizar acciones en los objetos de IBM MQ indicados en el mandato identificando a los usuarios por el nombre o revocando la autorización de todos los usuarios en *PUBLIC.

IBM i *21=Recuperar objeto*

Utilice Recuperar objeto para restaurar objetos dañados de la información almacenada en diarios de IBM MQ.

Recuperar objeto utiliza el mandato Volver a crear objeto MQ (RCRMQMOBJ) para recuperar todos los objetos dañados indicados en el mandato. Si un objeto no está dañado, no se realiza ninguna acción sobre dicho objeto.

IBM i *22=Registrar imagen*

Utilice Registrar imagen para reducir el número de receptores de diarios necesarios para la recuperación de un conjunto de objetos y minimizar el tiempo de recuperación.

El mandato RCDMQMIMG toma un punto de comprobación para todos los objetos que están seleccionados en el mandato. Sincroniza los valores actuales de los objetos en el sistema de archivos integrados (IFS) con la información más reciente sobre los objetos, tales como MQPUT y MQGET registrados en los receptores de diario.

Cuando el mandato que completa los objetos en el IFS está actualizado y ya no es necesario que estos receptores de diarios estén presentes para recuperar los objetos. Los receptores de diarios desconectados se pueden desconectar (siempre que no sea necesario que estén presentes para recuperar otros objetos).

IBM i Configuración de la comunicación para IBM i

Cuando se inicia un canal de gestión de colas distribuidas, éste intenta utilizar la conexión especificada en la definición de canal. Para que tenga éxito, es necesario que la conexión esté definida y disponible.

DQM es un recurso de gestión de colas remotas para IBM MQ for IBM i. Proporciona programas de control de canales para el gestor de colas de IBM MQ for IBM i que forman la interfaz con los enlaces de comunicación, controlables por el operador del sistema.

Cuando se inicia un canal de gestión de colas distribuidas, éste intenta utilizar la conexión especificada en la definición de canal. Para que tenga éxito, es necesario que la conexión esté definida y disponible. En esta sección se explica cómo asegurarse de que la conexión está definida y disponible.

Antes de que un canal pueda iniciarse, debe definirse la cola de transmisión como se describe en esta sección, y debe incluirse en la definición de canal de mensajes.

Puede elegir entre las dos formas de comunicación siguientes entre los sistemas IBM MQ for IBM i:

- “Definición de una conexión TCP en IBM i” en la página 302

  Para TCP, se puede utilizar una dirección de host, y estas conexiones se configuran como se describe en la publicación *IBM i Communication Configuration Reference*.

  En el entorno TCP, a cada servicio distribuido se le asigna una dirección TCP exclusiva que pueden utilizar las máquinas remotas para acceder al servicio. La dirección TCP consta de un nombre o número

de host y un número de puerto. Todos los gestores de colas utilizan ese número para comunicarse entre sí por medio de TCP.

- “Recepción en TCP” en la página 303

  Esta forma de comunicación exige la definición de una unidad lógica de tipo 6.2 (LU 6.2) de IBM i SNA que proporciona el enlace físico entre el sistema IBM i que da servicio al gestor de colas local y el sistema que da servicio al gestor de colas remoto. Consulte la publicación *IBM i Communication Configuration Reference* para obtener más detalles sobre la configuración de las comunicaciones en IBM i.

Cuando sea necesario, también debe estar preparada la disposición de desencadenante con la definición de los procesos y las colas necesarios.

MQ Adv. CD Un canal de mensajes que utiliza TCP/IP puede apuntar a un IBM Aspera faspio Gateway, que proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de la red. Un gestor de colas que se ejecuta en cualquier plataforma autorizada puede conectarse a través de un Aspera gateway. La propia pasarela se despliega en Red Hat o Ubuntu Linux, o Windows. Consulte Definición de una conexión de Aspera gateway en Linux o Windows.

Tareas relacionadas

“Supervisión y control de canales en IBM i” en la página 287
Utilice los mandatos y paneles de DQM para crear, supervisar y controlar los canales con gestores de colas remotos. Cada gestor de colas tiene un programa DQM para controlar las interconexiones con gestores de colas remotos compatibles.

Referencia relacionada

Configuración de ejemplo - IBM MQ for IBM i
Ejemplo de planificación de canal de mensajes para IBM MQ for IBM i
Trabajos de intercomunicación en IBM i
Estados de canal en IBM i

IBM i *Definición de una conexión TCP en IBM i*

Puede definir una conexión TCP dentro de la definición de canal utilizando el campo Nombre de conexión.

La definición de canal contiene un campo, NOMBRE DE CONEXIÓN, que contiene la dirección de red TCP del destino o el nombre de host (por ejemplo, ABCHOST). La dirección de red TCP puede estar en formato decimal separado por puntos IPv4 (por ejemplo, 127.0.0.1) o en formato hexadecimal IPv6 (por ejemplo, 2001:DB8:0:0:0:0:0:0). Si CONNECTION NAME es un nombre de host o un servidor de nombres, la tabla de host de IBM i se utiliza para convertir el nombre de host en una dirección de host TCP.

Para que la dirección TCP sea completa, hace falta un número de puerto; si no se proporciona, se utiliza el número de puerto predeterminado, 1414. En el extremo que inicia la conexión (tipos de canal emisor, peticionario y servidor) es posible ofrecer un número de puerto opcional para la conexión, por ejemplo:

```
Connection name 127.0.0.1 (1555)
```

En este caso, el extremo que inicia la conexión intenta conectarse a un programa receptor en el puerto 1555.

MQ Adv. CD Un canal de mensajes que utiliza TCP/IP puede apuntar a un IBM Aspera faspio Gateway, que proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de la red. Un gestor de colas que se ejecuta en cualquier plataforma autorizada puede conectarse a través de un Aspera gateway. La propia pasarela se despliega en Red Hat o Ubuntu Linux, o Windows. Consulte Definición de una conexión de Aspera gateway en Linux o Windows.

Utilización de la opción de reserva de escucha TCP

En TCP, las conexiones se tratan de forma incompleta a menos que tenga lugar un reconocimiento entre el servidor y el cliente. Estas conexiones se llaman solicitudes de conexión pendientes. Se establece

un valor máximo para estas solicitudes de conexión pendientes y se puede considerar una reserva de solicitudes en espera del puerto TCP para que el escucha acepte la solicitud.

Consulte “Utilización de la opción de proceso de escucha TCP en IBM MQ for Multiplatforms” en la página 284 para obtener más información y el valor específico para IBM i.

Conceptos relacionados

“Recepción en TCP” en la página 303
Los programas de canal receptor se inician en respuesta a una solicitud de inicio del canal emisor. Para responder a la solicitud de inicio, se debe iniciar un programa de escucha para detectar solicitudes de red entrantes e iniciar el canal asociado. Este programa de escucha se inicia con el mandato STRMQMLSR.

IBM i *Recepción en TCP*

Los programas de canal receptor se inician en respuesta a una solicitud de inicio del canal emisor. Para responder a la solicitud de inicio, se debe iniciar un programa de escucha para detectar solicitudes de red entrantes e iniciar el canal asociado. Este programa de escucha se inicia con el mandato STRMQMLSR.

Puede iniciar más de un escucha para cada gestor de colas. De forma predeterminada, el mandato STRMQMLSR utiliza el puerto 1414 pero puede alterar temporalmente este valor. Para alterar temporalmente el valor predeterminado, añada las sentencias siguientes al archivo qm.ini del gestor de colas seleccionado. En este ejemplo, es preciso que el escucha utilice el puerto 2500:

```
TCP:
Port=2500
```

El archivo qm.ini se encuentra en este directorio IFS: /QIBM/UserData/mqm/qmgrs/*nombre del gestor de colas*.

Este nuevo valor se lee sólo cuando se inicia el escucha TCP. Si tiene un escucha ya en ejecución, este cambio no lo ve dicho programa. Para utilizar el nuevo valor, detenga el escucha y vuelva a emitir el mandato STRMQMLSR. Ahora, siempre que utilice el mandato STRMQMLSR, el escucha toma, como valor predeterminado, el nuevo puerto.

Como alternativa, puede especificar un número de puerto diferente en el mandato STRMQMLSR. Por ejemplo:

```
STRMQMLSR MQMNAME( queue manager name ) PORT(2500)
```

Este cambio hace que el escuche tome, como valor predeterminado, el nuevo puerto mientras dure el trabajo del escucha.

Utilización de la opción TCP SO_KEEPALIVE

Si desea utilizar la opción SO_KEEPALIVE (si desea más información, consulte “Cómo comprobar que el otro extremo del canal sigue estando disponible” en la página 247) debe añadir la entrada siguiente al archivo de configuración del gestor de colas (qm.ini en el directorio IFS, /QIBM/UserData/mqm/qmgrs/*nombre de gestor de colas*):

```
TCP:
KeepAlive=yes
```

A continuación, debe emitir el mandato siguiente:

```
CFGTCP
```

Seleccione la opción 3 (Cambiar atributos TCP). Ahora puede especificar un intervalo de tiempo en minutos. Puede especificar un valor en el rango de 1 a 40320 minutos; el valor predeterminado es 120.

Utilización de la opción de reserva de escucha TCP

Al recibir en TCP, se establece un número máximo de solicitudes de conexión pendientes. Este número se puede considerar una *reserva* de solicitudes en espera del puerto TCP para que el escucha acepte la solicitud.

El valor de reserva del escucha predeterminado en IBM i es 255. Si la reserva alcanza este valor, la conexión TCP se rechaza y el canal no se puede iniciar.

En el caso de los canales MCA, el resultado es que el canal queda en estado de reintento (RETRY) y reintenta la conexión más adelante.

Para conexiones de cliente, el cliente recibe un código de razón MQRC_Q_MGR_NOT_AVAILABLE de MQCONN y puede reintentar la conexión en un momento posterior.

No obstante, para evitar este error, puede añadir una entrada en el archivo qm.ini:

```
ListenerBacklog = n
```

Esto altera temporalmente el número máximo predeterminado de solicitudes pendientes (255) para el escucha TCP.

Nota: Algunos sistemas operativos dan soporte a un valor mayor que el predeterminado. Si es necesario, este valor se puede utilizar para evitar alcanzar el límite de conexiones.

IBM i *Definición de una conexión LU 6.2 en IBM i*

Defina los detalles de las comunicaciones LU 6.2 utilizando un nombre de modalidad, nombre de TP y el nombre de una conexión LU 6.2 totalmente calificada.

El extremo iniciado del enlace debe tener una definición de entrada de direccionamiento para complementar este objeto CSI. Puede obtener más información sobre la gestión de solicitudes de trabajo de sistemas LU 6.2 remotos en la publicación *IBM i Programming: Work Management Guide*.

Para obtener más información, consulte la publicación *Multiplatform APPC Configuration Guide* y la tabla siguiente.

Tabla 26. Valores del sistema IBM i local para una plataforma de gestor de colas remoto

Plataforma remota	**TPNAME**
z/OS o MVS	El mismo que el de la información complementaria correspondiente sobre el gestor de colas remoto.
IBM i	El mismo que el valor de comparación de la entrada de direccionamiento del sistema IBM i.
Sistemas AIX and Linux	El programa de transacción invocable definido en la configuración de la conexión LU 6.2 remota.
Windows	El mismo que el especificado en el mandato Run Listener de Windows, o el programa de transacción invocable definido mediante TpSetup en Windows.

Si tiene más de un gestor de colas en el mismo sistema, asegúrese de que los TPnames de las definiciones de canal son exclusivos.

Conceptos relacionados

“Extremo de inicio (emisor)” en la página 305
Utilice el mandato CRTMQMCHL para definir un canal de transporte de tipo *LU62.

“Extremo iniciado (receptor)” en la página 307
Utilice el mandato CRTMQMCHL para definir el extremo receptor del enlace de canal de mensajes con tipo de transporte *LU62.

IBM i *Extremo de inicio (emisor)*

Utilice el mandato CRTMQMCHL para definir un canal de transporte de tipo *LU62.

El uso del objeto CSI es opcional en IBM MQ for IBM i V5.3 o posterior.

El panel del extremo de inicio aparece en la Panel de configuración de LU 6.2 - extremo de inicio. Para obtener el panel completo tal como se muestra, pulse F10 en el primer panel.

```
Create Comm Side Information (CRTCSI)

Type choices, press Enter.

Side information . . . . . > WINSDOA1   Name
Library . . . . . >  QSYS    Name, *CURLIB
Remote location . . . . . > WINSDOA1   Name
Transaction program . . . . . > MQSERIES

Text 'description' . . . . . .  *BLANK


Additional Parameters

Device . . . . . .  *LOC     Name, *LOC
Local location . . . . . .  *LOC     Name, *LOC, *NETATR
Mode . . . . . .  JSTMOD92   Name, *NETATR
Remote network identifier . . .  *LOC     Name, *LOC, *NETATR, *NONE
Authority . . . . . .  *LIBCRTAUT  Name, *LIBCRTAUT, *CHANGE...

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

Figura 34. Panel de configuración de comunicaciones de LU 6.2 - extremo de inicio

Complete los campos del extremo de inicio de la forma siguiente:

Información complementaria
Otorgue a esta definición un nombre que se utilice para almacenar el objeto de información complementaria que va a crear, por ejemplo, winsdoa1.

Nota: Para LU 6.2, el enlace entre la definición de canal de mensajes y la conexión de comunicación es el campo **Nombre de conexión** de la definición de canal de mensajes en el extremo emisor. Este campo contiene el nombre del objeto CSI.

Biblioteca
Nombre de la biblioteca donde se almacena esta definición.

El objeto CSI debe estar disponible en una biblioteca accesible al programa que sirve al canal de mensajes, por ejemplo, QSYS, QMQM y QGPL.

Si el nombre es incorrecto, falta o no se puede encontrar, se produce un error en el inicio del canal.

Ubicación remota
Especifica el nombre de ubicación remota con la que se comunica el programa.

En definitiva, este parámetro necesario contiene el nombre de unidad lógica del socio en el sistema remoto, tal y como se define en la descripción del dispositivo que se utiliza para el enlace de comunicación entre los dos sistemas.

El nombre de **Ubicación remota** se puede encontrar emitiendo el mandato DSPNETA en el sistema remoto y viendo el nombre de la ubicación local predeterminada.

Programa de transacciones
Especifica el nombre (de 64 caracteres) del programa de transacción en el sistema remoto que se va a iniciar. Puede ser un nombre de proceso de transacción, un nombre de programa, el nombre de canal o una serie de caracteres que coincide con el **Valor de comparación** en la entrada de direccionamiento.

Este parámetro es necesario.

Nota: Para especificar nombres de programas de transacción de servicio SNA, entre la representación hexadecimal del nombre del programa de transacción de servicio. Por ejemplo, para especificar un nombre de programa de transacción de servicio con una representación hexadecimal de 21F0F0F1, entraría X'21F0F0F1'.

Para obtener más información sobre nombres de programas de transacción de servicio SNA consulte la publicación *SNA Transaction Programmer's Reference* para LU Tipo 6.2.

Si el extremo receptor es otro sistema IBM i, el nombre del **Programa de transacción** se utiliza para hacer coincidir el objeto CSI en el extremo emisor con la entrada de direccionamiento en el extremo receptor. Este nombre debe ser exclusivo para cada gestor de colas en el sistema IBM i de destino. Consulte el parámetro **Programa a llamar** en Extremo iniciado (receptor). Consulte también el parámetro **Datos de comparación: valor de comparación** en el panel Añadir entrada de direccionamiento.

Texto descriptivo
Una descripción (de hasta 50 caracteres) para recordarle el uso previsto de esta conexión.

Dispositivo
Especifica el nombre de la descripción del dispositivo utilizada para el sistema remoto. Los valores posibles son:

***LOC**
El dispositivo está determinado por el sistema.

Nombre de dispositivo
Especifique el nombre del dispositivo que está asociado con la ubicación remota.

Ubicación Local
Especifica el nombre de la ubicación local. Los valores posibles son:

***LOC**
El nombre de la ubicación local está determinado por el sistema.

***NETATR**
El valor LCLLOCNAME especificado en los atributos de red del sistema.

Nombre de ubicación local
Especifique el nombre de la ubicación. Especifique la ubicación local si desea indicar un nombre de ubicación específico para la ubicación remota. El nombre de ubicación se puede encontrar mediante el mandato DSPNETA.

Modalidad
Especifica la modalidad utilizada para controlar la sesión. Este nombre es el mismo que la interfaz común de programación (CPI)- Mode_Name de comunicaciones. Los valores posibles son:

***NETATR**
Se utiliza la modalidad en los atributos de red.

BLANK
Se utilizan ocho caracteres en blanco.

Nombre de modalidad
Especifique un nombre de modalidad para la ubicación remota.

Nota: Dado que la modalidad determina la prioridad de transmisión de la sesión de comunicaciones, puede ser útil para definir las diferentes modalidades en función de la prioridad de los mensajes que se envían; por ejemplo MQMODE_HI, MQMODE_MED y MQMODE_LOW. (Puede tener más de un CSI que apunte a la misma ubicación.)

Identificador de red remota
Especifica el identificador de red remota utilizado con la ubicación remota. Los valores posibles son:

***LOC**
Se utiliza el ID de red remota para la ubicación remota.

***NETATR**
Se utiliza el identificador de red remota especificado en los atributos de red.

***NONE**
La red remota no tiene nombre.

ID de red remota
Especifique un ID de red remota. Utilice el mandato DSPNETA en la ubicación remota para encontrar el nombre de este ID de red. Es el ID de red local en la ubicación remota.

Autorización
Especifica la autorización que está otorgando a los usuarios que no tienen autorización específica sobre el objeto, que no están en una lista de autorizaciones y con un perfil de grupo que no tiene autorización específica sobre el objeto. Los valores posibles son:

***LIBCRTAUT**
La autorización pública sobre el objeto proviene del parámetro CRTAUT de la biblioteca especificada. Este valor se determina durante la creación. Si el valor CRTAUT para la biblioteca cambia después de crear el objeto, el valor nuevo no afecta a los objetos existentes.

***CHANGE**
Autorización de cambios permite al usuario realizar funciones básicas sobre el objeto; no obstante, el usuario no puede cambiar el objeto. Autorización de cambios proporciona autorización operativa sobre el objeto y toda la autorización de datos.

***ALL**
El usuario puede realizar todas las operaciones excepto aquellas limitadas al propietario o controladas por la autoridad de gestión de la lista de autorizaciones. El usuario puede controlar la existencia del objeto y especificar la seguridad del objeto, cambiar el objeto y realizar funciones básicas sobre el objeto. El usuario puede cambiar la propiedad del objeto.

***USE**
Autorización de uso proporciona autorización operativa sobre el objeto y autorización de lectura.

***EXCLUDE**
Autorización de exclusión impide al usuario acceder al objeto.

Lista de autorizaciones
Especifique el nombre de la lista de autorizaciones con autorización que se utiliza para la información complementaria.

IBM i *Extremo iniciado (receptor)*

Utilice el mandato CRTMQMCHL para definir el extremo receptor del enlace de canal de mensajes con tipo de transporte *LU62.

Deje el campo CONNECTION NAME en blanco y asegúrese de que los detalles correspondientes coinciden con el extremo emisor del canal. Para obtener información detallada, consulte Creación de un canal.

Para habilitar el extremo iniciado para que inicie el canal receptor, añada una entrada de direccionamiento a un subsistema en el extremo iniciado. El subsistema debe ser uno que asigna el dispositivo APPC utilizado en las sesiones LU 6.2. Por lo tanto, debe tener una entrada de comunicaciones válida para dicho dispositivo. La entrada de direccionamiento llama al programa que inicia el extremo receptor del canal de mensajes.

Utilice los mandatos IBM i (por ejemplo, ADDRTGE) para definir el extremo del enlace iniciado por una sesión de comunicación.

El panel de extremo iniciado se muestra en Panel de configuración de comunicaciones de LU 6.2 - añadir entrada de direccionamiento.

```
Add Routing Entry (ADDRTGE)

Type choices, press Enter.

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

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



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

Figura 35. Panel de configuración de comunicaciones de LU 6.2 - extremo iniciado

Descripción del subsistema
Nombre del subsistema donde reside esta definición. Utilice el mandato IBM i WRKSBSD para ver y actualizar la descripción del subsistema adecuado para la entrada de direccionamiento.

Número de secuencia de entrada de direccionamiento
Un número exclusivo del subsistema para identificar esta definición de comunicaciones. Puede utilizar valores en el rango de 1 - 9999.

Datos de comparación: valor de comparación
Una serie de texto para comparar con la serie recibida cuando se inicia la sesión mediante un parámetro **Programa de transacción**, tal como se muestra en la Figura 1. La serie de caracteres deriva del campo Programa de transacción del CSI del remitente.

Datos de comparación: Posición inicial
La posición del carácter en la serie en la que va a iniciarse la comparación.

Nota: El campo de posición inicial es la posición del carácter en la serie para la comparación y esta posición es siempre 37.

Programa a llamar
El nombre del programa que ejecuta el programa de mensajes de entrada que ha de llamarse para iniciar la sesión.

El programa, AMQCRC6A, se llama para el gestor de colas predeterminado. Este programa se suministra con IBM MQ for IBM i y establece el entorno y luego llama a AMQCRS6A.

Para gestores de colas adicionales:

- Cada gestor de colas tiene un programa LU 6.2 específico que se puede invocar y que se encuentra en la biblioteca. Este programa se denomina AMQCRC6B y se genera automáticamente cuando se crea el gestor de colas.
- Cada gestor de colas requiere una entrada de direccionamiento específica con datos exclusivos de direccionamiento que se añadirán. Estos datos de direccionamiento deben coincidir con el nombre del **Programa de transacción** que suministra el sistema solicitante (consulte Extremo de inicio (emisor)).

Se muestra un ejemplo en Panel de configuración de comunicaciones de LU 6.2 - visualizar entradas de direccionamiento:

```
Display Routing Entries
System:  MY400
Subsystem description:  QCMN      Status:  ACTIVE

Type options, press Enter.
5=Display details

Start
Opt  Seq Nbr  Program    Library    Compare Value   Pos
10   *RTGDTA           'QZSCSRVR'     37
20   *RTGDTA           'QZRCSRVR'     37
30   *RTGDTA           'QZHQTRG'     37
50   *RTGDTA           'QVPPRINT'     37
60   *RTGDTA           'QNPSERVR'     37
70   *RTGDTA           'QNMAPINGD'    37
80   QNMAREXECD  QSYS     'AREXECD'     37
90   AMQCRC6A   QMQMBW    'MQSERIES'     37
100   *RTGDTA           'QTFDWNLD'     37
150   *RTGDTA           'QMFRCVR'     37


F3=Exit  F9=Display all detailed descriptions  F12=Cancel
```

Figura 36. Panel de configuración de comunicaciones de LU 6.2 - extremo iniciado

En Panel de configuración de comunicaciones de LU 6.2 - visualizar entradas de direccionamiento, el número de secuencia 90 representa el gestor de colas predeterminado y proporciona compatibilidad con configuraciones de releases anteriores (es decir, V3R2, V3R6, V3R7 y V4R2) de IBM MQ for IBM i. Estos releases sólo permiten un gestor de colas. Los números de secuencia 92 y 94 representan dos gestores de colas adicionales denominados ALPHA y BETA que se crean con las bibliotecas QMALPHA y QMBETA.

Nota: Puede tener más de un entrada de direccionamiento para cada gestor de colas utilizando diferentes datos de direccionamiento. Estas entradas ofrecen la opción de diferentes prioridades de trabajo en función de las clases utilizadas.

Clase
El nombre y la biblioteca de la clase utilizada para los pasos iniciados a través de esta entrada de direccionamiento. La clase define los atributos del entorno de ejecución del paso de direccionamiento y especifica la prioridad del trabajo. Debe especificarse una entrada de clase adecuada. Utilice, por ejemplo, el mandato WRKCLS para visualizar clases existentes o para crear una clase. Puede obtener más información sobre la gestión de solicitudes de trabajo de sistemas LU 6.2 remotos en la publicación *IBM i Programming: Work Management Guide*.

Nota sobre la gestión de trabajo

El trabajo AMQCRS6A no puede aprovechar las características de gestión de trabajo normales de IBM i que se documentan en Gestión de trabajo, ya que no se inicia de la misma manera que otros trabajos de IBM MQ. Para cambiar las propiedades de ejecución de los trabajos receptores de LU62, puede realizar algunos de los cambios siguientes:

- Modifique la descripción de clase que se especifica en la entrada de direccionamiento para el trabajo AMQCRS6A
- Cambie la descripción de trabajo en la entrada de comunicaciones

Consulte *IBM iProgramación: Guía de gestión de trabajo* para obtener más información sobre la configuración de trabajos de comunicación.

Configuración de un clúster de gestores de colas

Los clústeres proporcionan un mecanismo para interconectar gestores de colas de forma que simplifica la configuración inicial y la gestión continua. Puede definir componentes de clúster, y crear y gestionar los clústeres.

Antes de empezar

Para obtener una introducción a los conceptos de agrupación en clúster, consulte Clústeres.

Cuando diseñe el clúster del gestor de colas, tendrá que tomar algunas decisiones. Consulte Clústeres de ejemplo y Diseño de clústeres.

Tareas relacionadas

"Mover una definición de tema de clúster a un gestor de colas diferente" en la página 463
Para los clústeres de direccionamiento de host de tema o de direccionamiento directo, es posible que necesite mover una definición de tema de clúster cuando anula un gestor de colas o cuando un gestor de colas del clúster falla o no está disponible durante un periodo de tiempo prolongado.

Referencia relacionada

DELETE TOPIC

Definición de componentes de un clúster

Los clústeres están formados por gestores de colas, canales de clúster y colas de clúster. Puede definir colas de clúster y modificar algunos aspectos de los objetos de clúster predeterminados. Puede obtener información acerca de la configuración y el estado de los canales definidos automáticamente, y acerca de la relación entre los canales de clúster emisor individuales y las colas de transmisión.

Consulte los subtemas siguientes para obtener más información sobre la definición de cada uno de los componentes del clúster:

Conceptos relacionados

Componentes de un clúster

Canales de clúster

Tareas relacionadas

Definición de temas de clúster

"Configurar un nuevo clúster" en la página 324
Siga estas instrucciones para configurar el clúster de ejemplo. Instrucciones separadas describen la configuración del clúster en TCP/IP, LU 6.2 y con una única cola de transmisión o varias colas de transmisión. Pruebe los trabajos del clúster enviando un mensaje de un gestor de colas a otro.

"Añadir un gestor de colas a un clúster" en la página 335
Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando la cola de transmisión de clúster única `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

Definición de cola de clúster

Una cola de clúster es una cola que se aloja en un gestor de colas de clúster y que está disponible para otros gestores de colas del clúster. Defina una cola de clúster como cola local en el gestor de colas de clúster donde está alojada la cola. Especifique el nombre del clúster al que pertenece la cola.

El ejemplo siguiente muestra un mandato **runmqsc** para definir una cola de clúster con la opción `CLUSTER`:

```
DEFINE QLOCAL(Q1) CLUSTER(SALES)
```

Una definición de cola de clúster se anuncia en otros gestores de colas del clúster. Los otros gestores de colas del clúster pueden transferir mensajes a una cola de clúster sin necesidad de que haya una definición de cola remota correspondiente. Una cola de clúster se puede anunciar en más de un clúster utilizando una lista de nombres de clúster.

Cuando se anuncia una cola, cualquier gestor de colas del clúster puede poner mensajes en ella. Para transferir un mensaje, el gestor de colas debe averiguar, en los repositorios completos, donde está alojada la cola. A continuación, añade información de direccionamiento al mensaje y pone el mensaje a una cola de transmisión de clúster.

Una cola de clúster puede ser una cola que se comparte entre miembros de un grupo de compartición de colas en IBM MQ for z/OS.

Enlazando

Puede crear un clúster en el que más de un gestor de colas aloje una instancia de la misma cola de clúster. Asegúrese de que todos los mensajes de una secuencia se envían a la misma instancia de la cola. Puede enlazar una serie de mensajes a una cola determinada utilizando la opción MQOO_BIND_ON_OPEN en la llamada MQOPEN.

Colas de transmisión de clúster

Un gestor de colas puede almacenar mensajes para otros gestores de colas en un clúster en varias colas de transmisión. Puede configurar un gestor de colas para almacenar mensajes en varias colas de transmisión de clúster de dos maneras diferentes. Si establece el atributo de gestor de colas **DEFCLXQ** en CHANNEL, se crea automáticamente una cola de transmisión de clúster diferente de SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE para cada canal de clúster emisor. Si establece la opción de cola de transmisión CLCHNAME para que coincida con uno o varios canales de clúster emisor, el gestor de colas puede almacenar mensajes para los canales coincidentes en esa cola de transmisión.

Atención: Si utiliza SYSTEM.CLUSTER.TRANSMIT.QUEUES dedicado con un gestor de colas que se ha actualizado desde una versión del producto anterior a IBM WebSphere MQ 7.5, asegúrese de que SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE tiene la opción SHARE/NOSHARE establecida en **SHARE**.

Un mensaje para una cola de clúster en un gestor de colas diferente se coloca en una cola de transmisión de clúster antes de enviarse. Una canal de clúster emisor transfiere los mensajes de una cola de transmisión de clúster a canales de clúster receptor en otros gestores de colas. De forma predeterminada, una cola de transmisión de clúster definida en el sistema conserva todos los mensajes que se van a transferir a otros gestores de colas de clúster. La cola se denomina SYSTEM.CLUSTER.TRANSMIT.QUEUE. Un gestor de colas que forma parte de un clúster puede enviar mensajes en esta cola de transmisión de clúster a cualquier otro gestor de colas en el mismo clúster.

De forma predeterminada, se crea una definición para la cola SYSTEM.CLUSTER.TRANSMIT.QUEUE única en cada gestor de colas Multiplatforms. En z/OS, la definición se puede definir con el **CSQ4INSX** de ejemplo suministrado.

Puede configurar un gestor de colas para transferir mensajes a otros gestores de colas en clúster utilizando varias colas de transmisión. Puede definir colas de transmisión de clúster adicionales manualmente o hacer que el gestor de colas las cree automáticamente.

Para hacer que el gestor de colas cree las colas automáticamente, cambie el atributo de gestor de colas DEFCLXQ de SCTQ a CHANNEL. El resultado es que el gestor de colas crea una cola de transmisión de clúster individual para cada canal de clúster emisor que se crea. Las colas de transmisión se crean como colas dinámicas permanentes desde la cola modelo, SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE. El nombre de cada cola dinámica permanente es SYSTEM.CLUSTER.TRANSMIT. *ChannelName*. El nombre del canal de clúster emisor con el que está asociada cada cola de transmisión de clúster dinámica permanente se establece en el atributo de cola de transmisión local CLCHNAME. Los mensajes para gestores de colas en clúster remotos se colocan en la cola de transmisión de clúster dinámica permanente para el canal de clúster emisor asociado, en lugar de hacerlo en SYSTEM.CLUSTER.TRANSMIT.QUEUE.

Para crear las colas de transmisión de clúster manualmente, cree una cola local con el atributo USAGE establecido en XMITQ, y el atributo CLCHNAME establecido en un nombre de canal genérico que se resuelva en uno o varios canales de clúster emisor; consulte ClusterChannelName. Si crea colas de transmisión de clúster manualmente, tiene la posibilidad de asociar la cola de transmisión con un solo canal de clúster emisor o con varios canales de clúster emisor. El atributo CLCHNAME es un nombre genérico que significa que puede colocar varios caracteres comodín "*" en el nombre.

Excepto para los canales de clúster emisor que se crean manualmente para conectar un gestor de colas a un repositorio completo, los canales de clúster emisor se crean automáticamente. Se crean

automáticamente cuando hay un mensaje para transferir a un gestor de colas de clúster. Se crean con el mismo nombre que el nombre del canal de clúster receptor que recibe mensajes de clúster para dicho clúster específico en el gestor de colas de destino.

Si sigue un convenio de denominación para canales de clúster receptor, es posible definir un valor genérico para CLCHNAME que filtre diferentes tipos de mensajes de clúster en diferentes colas de transmisión. Por ejemplo, si sigue el convenio de denominación para los canales de clúster receptor de *ClusterName. QmgrName*, el nombre genérico *ClusterName*.* filtra los mensajes para distintos clústeres en distintas colas de transmisión. Debe definir las colas de transmisión manualmente y establecer CLCHNAME en cada cola de transmisión en *ClusterName*.*.

Los cambios a la asociación de colas de transmisión de clúster en canales de clúster emisor no tienen efecto de forma inmediata. La cola de transmisión asociada actualmente a la que da servicio un canal de clúster emisor puede contener mensajes que están en el proceso de transferirse por el canal de clúster emisor. El gestor de colas puede cambiar la asociación de un canal de clúster emisor con una cola de transmisión diferente sólo cuando el canal de clúster emisor no procesa ningún mensaje en la cola de transmisión asociada en ese momento. Esto se puede producir cuando no hay ningún mensaje en la cola de transmisión para ser procesado por el canal de clúster emisor o cuando el proceso de mensajes queda suspendido y el canal de clúster emisor no tiene mensajes "en curso". Cuando esto sucede, los mensajes sin procesar para el canal de clúster emisor se transfieren a la cola de transmisión recién asociada y la asociación del canal de clúster emisor cambia.

Puede crear una definición de cola remota que se resuelva en una cola de transmisión de clúster. En la definición, el gestor de colas QMX está en el mismo clúster que el gestor de colas local, y no hay ninguna cola de transmisión, QMX.

```
DEFINE QREMOTE(A) RNAME(B) RQMNAME(QMX)
```

Durante la resolución de nombres de cola, la cola de transmisión de clúster tiene prioridad sobre la cola de transmisión predeterminada. Un mensaje transferido a A se almacena en la cola de transmisión de clúster y se envía a la cola remota B en QMX.

Los gestores de colas también pueden comunicarse con otros gestores de colas que no formen parte de un clúster. Debe definir canales y una cola de transmisión para el otro gestor de colas, del mismo modo que en un entorno de gestión de colas distribuidas.

Nota: Las aplicaciones deben grabar en colas que se resuelvan en la cola de transmisión de clúster, y no deben grabar directamente en la cola de transmisión de clúster.

Definición automática de colas remotas

Un gestor de colas de un clúster no necesita una definición de cola remota para las colas remotas del clúster. El gestor de colas de clúster localiza la ubicación de una cola remota en el repositorio completo. Añade información de direccionamiento al mensaje y lo transfiere a la cola de transmisión de clúster. IBM MQ crea automáticamente una definición equivalente a una definición de cola remota para que se pueda enviar el mensaje.

No se puede modificar ni suprimir una definición de cola remota creada automáticamente. No obstante, mediante el mandato **runmqsc** DISPLAY QUEUE con el atributo CLUSINFO, puede ver todas las colas locales de un gestor de colas, así como todas las colas de clúster, incluidas las colas de clúster en gestores de colas remotos. Por ejemplo:

```
DISPLAY QUEUE(*) CLUSINFO
```

Conceptos relacionados

Colas de clúster

Cómo seleccionar qué tipo de cola de transmisión de clúster se debe utilizar

Referencia relacionada

ClusterChannelName (MQCHAR20)

Trabajar con canales de clúster emisor definidos automáticamente

Después de introducir un gestor de colas en un clúster realizando sus definiciones CLUSSDR y CLUSRCVR iniciales, IBM MQ realiza automáticamente otras definiciones de canal de clúster emisor cuando es necesario para mover mensajes a otro gestor de colas del clúster. Puede ver información sobre los canales de clúster emisor definidos automáticamente pero no puede modificarlos. Para modificar su comportamiento, puede utilizar una salida de definición automática de canal.

Antes de empezar

Para obtener una introducción a los canales definidos automáticamente, consulte Canales de clúster emisor definidos automáticamente.

Acerca de esta tarea

Los canales de clúster emisor definidos automáticamente los crea el clúster cuando los necesita y permanecen activos hasta que se concluyen utilizando las reglas de intervalo de desconexión normales.

Los canales de remitente de clúster (CLUSSDR) se pueden definir automáticamente para mover mensajes de aplicación y mensajes de administración de clúster interno. Por ejemplo, en un Clúster de publicación/suscripción (uno en el que se ha definido un tema en clúster), se pueden definir canales entre repositorios parciales para permitir el intercambio de un estado de 'suscripción de proxy'. Cuando no se necesitan (están inactivos) durante un periodo de tiempo prolongado, los CLUSSDR autodefinidos se eliminan de la memoria caché de información del clúster de un repositorio parcial y dejan de ser visibles en ese gestor de colas.

Multi En Multiplatforms, el OAM (gestor de autorizaciones sobre objetos) no tiene conocimiento de la existencia de canales de clúster emisor definidos automáticamente. Si emite mandatos **start**, **stop**, **ping**, **reset** o **resolve** en un canal de clúster emisor definido automáticamente, el OAM comprueba si tiene autorización para realizar la misma acción en el canal de clúster receptor coincidente.

z/OS En z/OS, puede proteger un canal de clúster emisor definido automáticamente de la misma manera que cualquier otro canal.

Procedimiento

- Visualice información acerca de los canales definidos automáticamente para un gestor de colas del clúster concreto.

  No puede ver los canales definidos automáticamente utilizando el mandato DISPLAY CHANNEL **runmqsc** . Para ver los canales definidos automáticamente utilice el mandato siguiente:

  ```
  DISPLAY CLUSQMGR(qMgrName)
  ```

- Visualice el estado del canal definido automáticamente para un CLUSRCVR concreto.

  Para ver el estado del canal CLUSSDR definido automáticamente correspondiente a una definición de canal CLUSRCVR que ha creado, utilice el mandato siguiente:

  ```
  DISPLAY CHSTATUS(channelname)
  ```

- Utilice una salida de definición automática de canal para modificar el comportamiento de un canal definido automáticamente.

  Puede utilizar la salida de definición automática de canal de IBM MQ si desea escribir un programa de salida de usuario para personalizar un canal de clúster emisor o un canal de clúster receptor. Por ejemplo, puede utilizar la salida de definición automática de canal en un entorno de clúster para realizar cualquiera de las modificaciones siguientes:

  - Personalizar definiciones de comunicaciones, es decir, nombres de SNA LU 6.2.

- Añadir o eliminar otras salidas, por ejemplo, salidas de seguridad.
- Cambiar los nombres de salidas de canal.

  El nombre de la salida de canal CLUSSDR se genera automáticamente a partir de la definición de canal CLUSRCVR y, por lo tanto, es posible que no sea adecuada a sus necesidades, sobretodo si los dos extremos del canal están plataformas diferentes.

  El formato de los nombres de salida es distinto en plataformas diferentes. Por ejemplo:

  - z/OS En la plataforma z/OS, el formato del parámetro SCYEXIT (*nombre salida de seguridad*) es `SCYEXIT('SECEXIT')`
  - Windows En las plataformas Windows, el formato del parámetro SCYEXIT (*nombre salida de seguridad*) es `SCYEXIT(' drive:\path\library (secexit)')`

  Nota: z/OS Si no hay ninguna salida de definición automática de canal, el gestor de colas de z/OS obtiene el nombre de salida de canal CLUSSDR de la definición de canal CLUSRCVR en el otro extremo del canal. Para obtener el nombre de salida de z/OS de un nombre no de z/OS, se utiliza el siguiente algoritmo:

  - Los nombres de salida en Multiplatforms tienen el formato general *vía_acceso*(*biblioteca*(*función*).
  - Si *función* está presente, se utilizan hasta ocho caracteres de esa función.
  - De lo contrario, se utilizan hasta ocho caracteres de *biblioteca*.

  Por ejemplo:

  - `/var/mqm/exits/myExit.so(MsgExit)` se convierte en `MSGEXIT`
  - `/var/mqm/exits/myExit` se convierte en `MYEXIT`
  - `/var/mqm/exits/myExit.so(ExitLongName)` se convierte en `EXITLONG`
- Si el clúster necesita utilizar **PROPCTL** para eliminar cabeceras de aplicación como RFH2 de los mensajes que van de un gestor de colas de IBM MQ a un gestor de colas en una versión anterior del producto, debe escribir una salida de definición automática de canal que establezca **PROPCTL** en un valor de NONE.

- Utilice el atributo de canal LOCLADDR para controlar aspectos de direccionamiento.
  - Para permitir que un canal de salida (TCP) utilice una dirección IP, un puerto o un rango de puertos específico, utilice el atributo de canal LOCLADDR. Esta opción es útil si tiene más de una tarjeta de red y desea que un canal utilice una específica para las comunicaciones de salida.
  - Para especificar una dirección IP virtual en canales CLUSSDR, utilice la dirección IP del atributo LOCLADDR en un CLUSSDR definido manualmente. Para especificar el rango de puertos, utilice el rango de puertos del CLUSRCVR.
  - Si un clúster necesita utilizar LOCLADDR para obtener los canales de comunicación de salida para enlazar con una dirección IP específica, puede escribir una salida de definición automática de canal para forzar el valor LOCLADDR en cualquiera de sus canales CLUSSDR definidos automáticamente. También debe especificarlo en el canal CLUSSDR definido manualmente.
  - Especifique un número de puerto o un rango de puertos en el campo LOCLADDR de un canal CLUSRCVR, si desea que todos los gestores de colas de un clúster utilicen un puerto o un rango de puertos específico, para todas las comunicaciones de salida.

  Nota: No especifique una dirección IP en el campo LOCLADDR de un canal CLUSRCVR, a menos que todos los gestores de colas estén en el mismo servidor. La dirección IP de LOCLADDR se propaga a los canales CLUSSDR definidos automáticamente de todos los gestores de colas que se conectan utilizando el canal CLUSRCVR.

  Multi En Multiplatforms, puede establecer un valor de dirección local predeterminado que se utilizará para todos los canales emisores que no tienen una dirección local definida. El valor

predeterminado se define estableciendo la variable de entorno MQ_LCLADDR antes de iniciar el gestor de colas. El formato del valor coincide con el del atributo MQSC LOCLADDR.

Referencia relacionada

Dirección local (LOCLADDR)

Cómo trabajar con objetos de clúster predeterminado

Puede modificar las definiciones de canal predeterminadas igual que cualquier otra definición de canal, ejecutando mandatos MQSC o PCF. No altere las definiciones de cola predeterminadas, excepto SYSTEM.CLUSTER.HISTORY.QUEUE.

Para obtener una lista completa de estos objetos, consulte Objetos de clúster predeterminados. La lista siguiente sólo incluye los objetos que puede cambiar.

SYSTEM.CLUSTER.HISTORY.QUEUE

Cada gestor de colas de un clúster tiene una cola local denominada SYSTEM.CLUSTER.HISTORY.QUEUE. SYSTEM.CLUSTER.HISTORY.QUEUE se utiliza para almacenar el historial de información de estado de clúster para fines de servicio.

En los valores de objeto predeterminados, SYSTEM.CLUSTER.HISTORY.QUEUE se establece en PUT (ENABLED). Para suprimir la recopilación de historial, cambie el valor a PUT (DISABLED).

SYSTEM.CLUSTER.TRANSMIT.QUEUE

Cada gestor de colas tiene una definición para una cola local denominada SYSTEM.CLUSTER.TRANSMIT.QUEUE. SYSTEM.CLUSTER.TRANSMIT.QUEUE es la cola de transmisión predeterminada para todos los mensajes a todas las colas y gestores de colas que están dentro de clústeres. Puede cambiar la cola de transmisión predeterminada para cada canal de clúster emisor a SYSTEM.CLUSTER.TRANSMIT.*ChannelName*, cambiando el atributo del gestor de colas DEFXMITQ z/OS, excepto en z/OS.
No puede suprimirSYSTEM.CLUSTER.TRANSMIT.QUEUE. También se utiliza para definir comprobaciones de autorización si la cola de transmisión predeterminada que se utiliza es SYSTEM.CLUSTER.TRANSMIT.QUEUE o SYSTEM.CLUSTER.TRANSMIT.*ChannelName*.

Conceptos relacionados

Objetos de clúster predeterminado

Cómo trabajar con colas de transmisión de clúster y canales de clúster emisor

Los mensajes entre gestores de colas en clúster se almacenan en colas de transmisión de clúster y se reenvían por canales de clúster emisor. En cualquier momento, un canal de clúster emisor está asociado a una sola cola de transmisión. Si cambia la configuración del canal, éste puede cambiar a una cola de transmisión diferente la próxima vez que se inicie. El proceso de este conmutador está automatizado y es transaccional.

Ejecute el siguiente mandato MQSC para ver las colas de transmisión a las que están asociados los canales de clúster emisor:

```
DISPLAY CHSTATUS(*) WHERE(CHLTYPE EQ CLUSSDR)
```

```
AMQ8417: Display Channel Status details.
CHANNEL(TO.QM2)          CHLTYPE(CLUSSDR)
CONNAME(9.146.163.190(1416))    CURRENT
RQMNAME(QM2)             STATUS(STOPPED)
SUBSTATE( )             XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

La cola de transmisión de un canal de clúster emisor en estado detenido puede cambiar cuando el canal se vuelve a iniciar. “Selección de colas de transmisión predeterminadas por canales de clúster emisor” en la página 316 describe el proceso de selección de una cola de transmisión predeterminada; “Selección de colas de transmisión definidas manualmente por canales de clúster emisor” en la página 317 describe el proceso de selección de una cola de transmisión definida manualmente.

Cuando se inicia un canal de clúster emisor, comprueba su asociación con colas de transmisión. Si la configuración de las colas de transmisión cambia, o los valores predeterminados del gestor de colas cambian, el canal se puede reasociar a una cola de transmisión diferente. Si el canal se reinicia con una cola de transmisión diferente como resultado de un cambio de configuración, tiene lugar una transferencia de mensajes a la cola de transmisión recién asociada. "Cómo funciona el proceso de conmutación de un canal de clúster emisor a una cola de transmisión diferente" en la página 318 describe la transferencia de un canal de clúster emisor desde una cola de transmisión a otra.

El comportamiento de los canales de clúster emisor es distinto al de los canales emisores y canales servidores. Permanecen asociados a la misma cola de transmisión hasta que se modifique el atributo de canal **XMITQ**. Si modifica el atributo de cola de transmisión en un canal emisor o canal servidor y reinicia el canal, los mensajes no se transfieren desde la cola de transmisión antigua a la nueva.

Otra diferencia entre los canales de clúster emisor y los canales emisores o servidores es que varios canales de clúster emisor pueden abrir una cola de transmisión de clúster, pero sólo un canal emisor o canal servidor puede abrir una cola de transmisión normal. Tiene la opción de canales de clúster emisor que no comparten colas de transmisión. No se aplica una regla de exclusividad, sino que es un resultado de la configuración. Puede configurar la ruta que un mensaje sigue en un clúster para que no comparta colas de transmisión ni canales con mensajes que fluyen entre otras aplicaciones. Consulte Agrupación en clúster: Planificación de cómo configurar las colas de transmisión de clúster y "Añadir un clúster y una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela" en la página 368.

Selección de colas de transmisión predeterminadas por canales de clúster emisor

Una cola de transmisión de clúster es una cola predeterminada del sistema, con un nombre que empieza por SYSTEM.CLUSTER.TRANSMIT, o una cola definida manualmente. Un canal de clúster emisor se asocia a una cola de transmisión de clúster de una de estas dos formas: mediante la cola de transmisión de clúster predeterminada, o mediante configuración manual.

La cola de transmisión de clúster predeterminada se define como un atributo del gestor de colas, **DEFCLXQ**. Su valor es SCTQ o CHANNEL. El valor se establece en SCTQ para los gestores de colas nuevos y migrados. Puede cambiar el valor a CHANNEL.

Si el valor está establecido en SCTQ, la cola de transmisión de clúster predeterminada es SYSTEM.CLUSTER.TRANSMIT.QUEUE. Cualquier canal de clúster emisor puede abrir esta cola. Los canales de clúster emisor que abren la cola son los que no están asociados a colas de transmisión de clúster definidas manualmente.

Si se establece el valor CHANNEL, el gestor de colas puede crear una cola de transmisión dinámica permanente separada para cada canal de clúster emisor. Cada cola se denomina SYSTEM.CLUSTER.TRANSMIT. *ChannelName* y se crea a partir de la cola modelo, SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE. Cada canal de clúster emisor que no está asociado a una cola de transmisión de clúster definida manualmente se asocia a una cola de transmisión de clúster dinámica permanente. La cola la crea el gestor de colas cuando necesita una cola de transmisión de clúster independiente para el destino de clúster atendido por este canal de clúster emisor, y no existe ninguna cola.

Algunos destinos de clúster puede ser atendidos por canales de clúster emisor asociados a colas de transmisión definidas manualmente, y otros por la cola o colas predeterminadas. En la asociación de canales de clúster emisor con colas de transmisión, las colas de transmisión definidas manualmente siempre tienen prioridad sobre las colas de transmisión predeterminadas.

La Figura 37 en la página 317 muestra el orden de prioridad de las colas de transmisión de clúster. El único canal de clúster emisor no asociado a una cola de transmisión de clúster definida manualmente es CS.QM1. No está asociado a una cola de transmisión definida manualmente porque ninguno de los nombres de canal contenidos en el atributo **CLCHNAME** de las colas de transmisión coincide con CS.QM1.

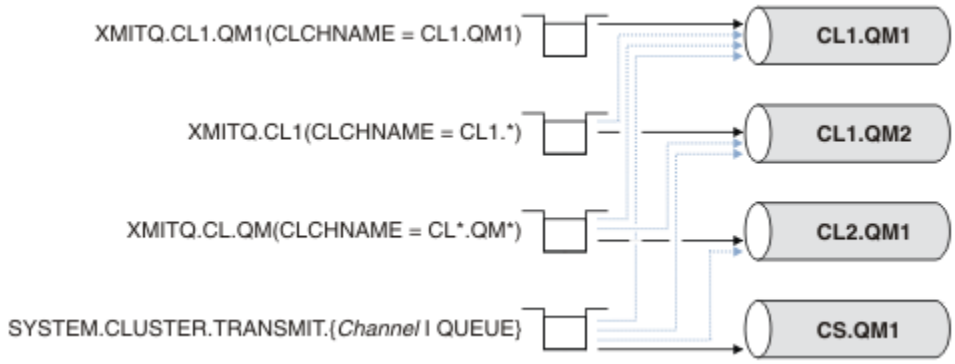


Figura 37. Prioridad de cola de transmisión / canal de clúster emisor

Selección de colas de transmisión definidas manualmente por canales de clúster emisor

Una cola definida manualmente tiene el atributo de cola de transmisión **USAGE** establecido en XMITQ, y el atributo de nombre de canal de clúster **CLCHNAME** establecido en un nombre de canal específico o genérico.

Si el nombre contenido en el atributo de cola **CLCHNAME** coincide con un nombre de canal de clúster emisor, el canal se asocia a la cola. El nombre puede ser una coincidencia exacta si el nombre no contiene caracteres comodín, o la mejor coincidencia si el nombre contiene caracteres comodín.

Si hay definiciones **CLCHNAME** en varias colas de transmisión que coinciden con el mismo canal de clúster emisor, se dice que las definiciones se solapan. Para resolver la ambigüedad, hay un orden de prioridad entre las coincidencias. Las coincidencias exactas siempre tienen prioridad. La Figura 37 en la página 317 muestra las asociaciones entre colas de transmisión y canales de clúster emisor. Las flechas negras muestran asociaciones actuales, y las flechas grises, asociaciones posibles. El orden de prioridad de las colas de transmisión en la Figura 37 en la página 317 es,

XMITQ.CL1.QM1
: La cola de transmisión XMITQ.CL1.QM1 tiene su atributo **CLCHNAME** establecido en CL1.QM1. La definición del atributo **CLCHNAME**, CL1.QM1, no tiene comodines, y tiene prioridad sobre cualquier otro atributo CLCHNAME, definido en otras colas de transmisión, que coincida con comodines. El gestor de colas almacena cualquier mensaje de clúster que el canal de clúster emisor CL1.QM1 deba transferir en la cola de transmisión XMITQ.CL1.QM1. La única excepción es si varias colas de transmisión tienen su atributo **CLCHNAME** establecido en CL1.QM1. En ese caso, el gestor de colas almacena los mensajes para el canal de clúster emisor CL1.QM1 en cualquiera de dichas colas. Selecciona una cola de forma arbitraria cuando se inicia el canal. Es posible que seleccione una cola diferente cuando el canal se inicie de nuevo.

XMITQ.CL1
: La cola de transmisión XMITQ.CL1 tiene su atributo **CLCHNAME** establecido en CL1.*. La definición del atributo **CLCHNAME**, CL1.*, tiene un comodín final, que coincide con el nombre de cualquier canal de clúster emisor que comience con CL1.. El gestor de colas almacena cualquier mensaje de clúster que se vaya a transferir mediante cualquier canal de clúster emisor cuyo nombre empiece por CL1. en la cola de transmisión XMITQ.CL1, a menos que haya una cola de transmisión con una coincidencia más específica, como por ejemplo la cola XMITQ.CL1.QM1. Un carácter comodín final hace que la definición sea menos específica que una definición sin comodines, y más específica que una definición con varios comodines, o comodines que van seguidos de más caracteres de cola.

XMITQ.CL.QM
: XMITQ.CL.QM es el nombre de la cola de transmisión con su atributo **CLCHNAME** establecido en CL*.QM*. La definición de CL*.QM* tiene dos comodines, que coinciden con el nombre de cualquier canal de clúster emisor que comience con CL., y que incluya o termine por QM. La coincidencia es menos específica que una coincidencia con un comodín.

SYSTEM.CLUSTER.TRANSMIT. *channelName*|QUEUE
Si ninguna cola de transmisión tiene un atributo **CLCHNAME** que coincida con el nombre del canal de clúster emisor que el gestor de colas va a utilizar, entonces el gestor de colas utiliza la cola de transmisión de clúster predeterminada. La cola de transmisión de clúster predeterminada es la única cola de transmisión de clúster del sistema, SYSTEM.CLUSTER.TRANSMIT.QUEUE, o una cola de transmisión de clúster del sistema que el gestor de colas ha creado para un canal de clúster emisor específico, SYSTEM.CLUSTER.TRANSMIT. *channelName*. Cuál de las colas es la predeterminada depende del valor del atributo de gestor de colas **DEFXMITQ**.

Consejo: A menos que tenga una clara necesidad de definiciones solapadas, procure evitarlas, ya que pueden dar lugar a configuraciones complejas que sean difíciles de comprender.

Cómo funciona el proceso de conmutación de un canal de clúster emisor a una cola de transmisión diferente

Para cambiar la asociación de canales de clúster emisor a colas de transmisión de clúster, cambie el parámetro **CLCHNAME** de cualquier cola de transmisión o el parámetro del gestor de colas **DEFCLXQ** cuando desee. No ocurre nada inmediatamente. Los cambios sólo se producen cuando se inicia un canal. Cuando se inicia, el canal comprueba si debe continuar reenviando mensajes desde la misma cola de transmisión. Existen tres tipos de cambio que modifican la asociación de un canal de clúster emisor con una cola de transmisión.

1. Redefinir el parámetro **CLCHNAME** de la cola de transmisión a la que está asociado actualmente del canal de clúster emisor para que sea menos específico o dejarlo en blanco, o suprimir la cola de transmisión de clúster cuando el canal esté detenido.

   Ahora alguna otra cola de transmisión de clúster puede ahora coincidir mejor con el nombre de canal. O, si no ninguna otra cola de transmisión coincide con el nombre del canal de clúster emisor, la asociación debe revertir a la cola de transmisión predeterminada.
2. Redefinir el parámetro **CLCHNAME** de cualquier otra cola de transmisión de clúster, o añadir una cola de transmisión de clúster.

   El parámetro **CLCHNAME** de otra cola de transmisión puede ahora coincidir mejor con el canal de clúster emisor que la cola de transmisión a la que está asociada actualmente el canal de clúster emisor. Si el canal de clúster emisor está asociado actualmente a una cola de transmisión de clúster predeterminada, puede pasar a estar asociado a una cola de transmisión de clúster definida manualmente.
3. Si el canal de clúster emisor está asociado actualmente a una cola de transmisión de clúster predeterminada, cambiar el parámetro del gestor de colas **DEFCLXQ**.

Si la asociación de un canal de clúster emisor cambia, cuando el canal se inicia, conmuta su asociación a la nueva cola de transmisión. Durante la conmutación, se asegura de que no se pierda ningún mensaje. Los mensajes se transfieren a la nueva cola de transmisión en el orden en que el canal transferiría los mensajes al gestor de colas remoto.

Recuerde: Así como cualquier reenvío de mensajes en un clúster, debe poner mensajes en grupos para asegurarse de que los mensajes que deben entregarse en orden se entreguen en orden. En raras ocasiones, los mensajes pueden quedar desordenados en un clúster.

El proceso de conmutación pasa por los siguientes pasos transaccionales. Si el proceso de conmutación se interrumpe, el paso transaccional actual se reanuda cuando el canal se reinicia de nuevo.

Paso 1 - Procesar mensajes de la cola de transmisión original
El canal de clúster emisor se asocia a la nueva cola de transmisión, que puede compartir con otros canales de clúster emisor. Los mensajes destinados al canal de clúster emisor se siguen colocando en la cola de transmisión original. Un proceso de conmutación de transición transfiere mensajes desde la cola de transmisión original a la nueva cola de transmisión. El canal de clúster emisor reenvía los mensajes desde la nueva cola de transmisión al canal de clúster receptor. El estado del canal muestra que el canal de clúster emisor está todavía asociado a la cola de transmisión antigua.

El proceso de conmutación continúa transfiriendo mensajes recién llegados también. Este paso continúa hasta que el número de mensajes restantes que deben ser reenviados por el proceso de conmutación llega a cero. Cuando el número de mensajes llega a cero, el procedimiento pasa al paso 2.

Durante el paso 1, la actividad de disco para el canal aumenta. Los mensajes persistentes se confirman desde la primera cola de transmisión y en la segunda cola de transmisión. Esta actividad de disco es adicional a la que se realiza en los mensajes que se confirman cuando se colocan en la cola de transmisión y se eliminan de ella como parte de la transferencia normal de los mensajes. Idealmente, no llegan mensajes durante el proceso de conmutación, por lo que la transición puede tener lugar tan rápidamente como sea posible. Si llegan mensajes, son procesados por el proceso de conmutación.

Paso 2 - Procesar mensajes de la cola de transmisión nueva

Tan pronto como no queden mensajes en la cola de transmisión original para el canal de clúster emisor, los nuevos mensajes se colocan directamente en la cola de transmisión nueva. El estado del canal muestra que el canal de clúster emisor está asociado a la cola de transmisión nueva. El mensaje siguiente se escribe en el registro de errores del gestor de colas: " AMQ7341 `la cola de transmisión para el canal` *`ChannelName`* `es` *`QueueName`* ."

Varias colas de transmisión de clúster y atributos de cola de transmisión de clúster

Tiene la opción de reenviar mensajes de clúster a diferentes gestores de colas almacenando los mensajes en una sola cola de transmisión de clúster, o varias colas. Cuando utiliza una sola cola, tiene un solo conjunto de atributos de cola de transmisión de clúster para definir y consultar; cuando utiliza varias colas, tiene varios conjuntos de atributos. Para algunos atributos, tener varios conjuntos es una ventaja: por ejemplo, consultar la profundidad de la cola le indica cuántos mensajes están a la espera de ser reenviados por uno o varios canales, en lugar de por todos los canales. Para otros atributos, tener varios conjuntos es una desventaja: por ejemplo, probablemente no desea configurar los mismos permisos de acceso para cada cola de transmisión de clúster. Por este motivo, los permisos de acceso siempre se validan por comparación con el perfil de `SYSTEM.CLUSTER.TRANSMIT.QUEUE`, y no con los perfiles de una cola de transmisión de clúster determinada. Si desea aplicar controles de seguridad más detallados, consulte Control de accesos y varias colas de transmisión de clúster.

Varios canales de clúster emisor y varias colas de transmisión

Un gestor de colas almacena un mensaje en una cola de transmisión de clúster antes de reenviarlo en un canal de clúster emisor. El gestor de colas selecciona un canal de clúster emisor que está conectado al destino para el mensaje. El gestor puede disponer de una gama de canales de clúster emisor que todos se conectan al mismo destino. El destino puede ser la misma cola física, conectada por varios canales de clúster emisor a un solo gestor de colas. El destino puede ser también muchas colas físicas con el mismo nombre de cola, alojadas en gestores de colas diferentes del mismo clúster. Cuando existen varios canales de clúster emisores conectados a un destino, el algoritmo de equilibrio de la carga de trabajo elige uno. La elección depende de varios factores; consulte Algoritmo de gestión de la carga de trabajo del clúster.

En Figura 38 en la página 320, `CL1.QM1`, `CL1.QM2` y `CS.QM1` son todos ellos canales que podrían conducir al mismo destino. Por ejemplo, si define `Q1` en `CL1` en `QM1` y `QM2`, `CL1.QM1` y `CL1.QM2` proporcionan rutas hacia el mismo destino, `Q1`, en dos gestores de colas diferentes. Si el canal `CS.QM1` también está en `CL1`, también es un canal que puede ser utilizado por un mensaje destinado a `Q1`. La pertenencia al clúster de `CS.QM1` podría estar definida por una lista de nombres de clúster, que es la razón por la que el nombre de canal no incluye un nombre de clúster en su construcción. Dependiendo de los parámetros de equilibrio de la carga de trabajo y de la aplicación emisora, algunos mensajes destinados a `Q1` pueden ser colocados en cada una de las colas de transmisión, `XMITQ.CL1.QM1`, `XMITQ.CL1` y `SYSTEM.CLUSTER.TRANSMIT.CS.QM1`.

Si piensa separar el tráfico de mensajes para que los mensajes con el mismo destino no compartan colas ni canales con mensajes con destinos diferentes, puede primero dividir el tráfico en diferentes canales de clúster emisor y luego separar los mensajes para un canal determinado en una cola de transmisión diferente. Las colas de clúster del mismo clúster, en el mismo gestor de colas, normalmente

comparten los mismos canales de clúster. Definir varias colas de transmisión de clúster no es suficiente por sí solo para separar el tráfico de mensajes de clúster en colas diferentes. A menos que separe los mensajes destinados a colas diferentes en canales diferentes, los mensajes comparten la misma cola de transmisión de clúster.

Una forma sencilla de separar los canales utilizados por los mensajes es crear varios clústeres. En cualquier gestor de colas de cada clúster, defina una sola cola de clúster. A continuación, si define un canal de clúster receptor diferente para cada combinación de clúster/gestor de colas, los mensajes para cada cola de clúster no comparten un canal de clúster con los mensajes para otras colas de clúster. Si define colas de transmisión separadas para los canales de clúster, el gestor de colas emisor almacena mensajes para una sola cola de clúster en cada cola de transmisión. Por ejemplo, si desea que dos colas de clúster no compartan recursos, puede colocarlos en clústeres diferentes del mismo gestor de colas, o en gestores de colas diferentes del mismo clúster.

La elección de la cola de transmisión de clúster no afecta al algoritmo de equilibrio de la carga de trabajo. El algoritmo de equilibrio de la carga de trabajo elige el canal de clúster emisor que debe reenviar un mensaje. Coloca el mensaje en la cola de transmisión que es atendida por ese canal. Si se invoca de nuevo el algoritmo de equilibrio de la carga de trabajo, por ejemplo, cuando el canal se detiene, el algoritmo puede elegir un canal diferente para reenviar el mensaje. Si el algoritmo elige un canal diferente, y el nuevo canal reenvía mensajes desde una cola de transmisión de clúster diferente, el algoritmo transfiere el mensaje a la otra cola de transmisión.

En Figura 38 en la página 320, dos canales de clúster emisor, `CS.QM1` y `CS.QM2`, están asociados a la cola de transmisión predeterminada del sistema. Cuando el algoritmo de equilibrio de la carga de trabajo almacena un mensaje en `SYSTEM.CLUSTER.TRANSMIT.QUEUE`, o cualquier otra cola de transmisión de clúster, el nombre del canal de clúster emisor que debe reenviar el mensaje se almacena en el ID de correlación del mensaje. Cada canal sólo reenvía los mensajes para los que el ID de correlación coincide con el nombre de canal.

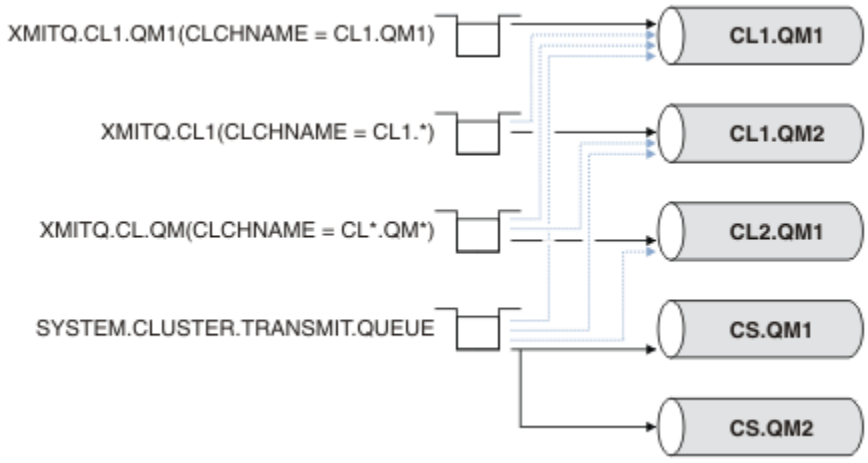


Figura 38. Varios canales de clúster emisor

Si `CS.QM1` se detiene, se examinan los mensajes de la cola de transmisión correspondiente a ese canal de clúster emisor. Los mensajes que se pueden reenviar por otro canal vuelven a ser procesados por el algoritmo de equilibrio de la carga de trabajo. Su ID de correlación se establece en un nombre alternativo de canal de clúster emisor. Si el canal alternativo es `CS.QM2`, el mensaje permanece en `SYSTEM.CLUSTER.TRANSMIT.QUEUE`. Si el canal alternativo es `CL1.QM1`, el algoritmo de equilibrio de la carga de trabajo transfiere el mensaje a `XMITQ.CL1.QM1`. Cuando el canal de clúster emisor se reinicia, los nuevos mensajes, y los mensajes que no se han marcado para otro canal de clúster emisor, se transfieren por el canal de nuevo.

Puede cambiar la asociación entre las colas de transmisión y los canales de clúster emisor en un sistema en ejecución. Puede cambiar un parámetro **CLCHNAME** en una cola de transmisión, o cambiar el parámetro del gestor de colas **DEFCLXQ**. Cuando se reinicia un canal que está afectado por el cambio, se inicia la conmutación de la cola de transmisión; consulte "Cómo funciona el proceso de conmutación de un canal de clúster emisor a una cola de transmisión diferente" en la página 318.

El proceso para conmutar la cola de transmisión comienza cuando se reinicia el canal. El reequilibrio de la carga de trabajo se inicia cuando se detiene el canal. Los dos procesos se pueden ejecutar en paralelo.

En el caso simple, la detención de un canal de clúster emisor no hace que el proceso de reequilibrio cambie el canal de clúster emisor que debe reenviar los mensajes de la cola. Esta situación se produce cuando no hay ningún otro canal de clúster emisor que pueda reenviar los mensajes al destino correcto. En este caso, los mensajes permanecen asignados al mismo canal de clúster emisor cuando éste se detiene. Cuando se inicia el canal, si hay un proceso de conmutación pendiente, el proceso de conmutación mueve los mensajes a una cola de transmisión diferente, donde los mensajes son procesados por el mismo canal de clúster emisor.

En el caso más complejo, existe más de un canal de clúster emisor que puede enviar algunos mensajes hacia el mismo destino. Puede detener y reiniciar el canal de clúster emisor para desencadenar la conmutación de la cola de transmisión. En muchos casos, cuando se reinicia el canal, el algoritmo de equilibrio de la carga de trabajo ya ha trasladado mensajes desde la cola de transmisión original a otras colas de transmisión atendidas por canales de clúster emisor diferentes. Sólo aquellos mensajes que no se pueden reenviar mediante un canal de clúster emisor diferente quedan pendientes de ser transferidos a la nueva cola de transmisión. En algunos casos, si el canal se reinicia rápidamente, algunos mensajes que podrían ser transferidos por el algoritmo de equilibrio de la carga de trabajo permanecen. En cuyo caso, algunos mensajes restantes son conmutados por el proceso de equilibrio de la carga de trabajo, y algunos por el proceso de conmutación de la cola de transmisión.

Conceptos relacionados

Canales de clúster

Agrupación en clúster: Aislamiento de aplicaciones utilizando varias colas de transmisión de clúster

"Cálculo del tamaño del registro" en la página 680
Cálculo del tamaño de las anotaciones cronológicas que un gestor de colas necesita.

Tareas relacionadas

Agrupación en clúster: Planificación de cómo configurar las colas de transmisión de clúster

"Creación de dos clústeres solapados con un gestor de cola de pasarela" en la página 358
Siga las instrucciones de la tarea para crear clústeres solapados con un gestor de colas de pasarela. Utilice los clústeres como punto de inicio para los siguientes ejemplos de aislamiento de mensajes dirigidos a una aplicación de los mensajes dirigidos a otras aplicaciones de un clúster.

"Añadir un gestor de colas a un clúster: colas de transmisión separadas" en la página 337
Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando varias colas de transmisión de clúster.

"Añadir una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela" en la página 365
Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza una cola de transmisión de clúster adicional para separar el tráfico de mensajes a un único gestor de colas de un clúster.

"Añadir un clúster y una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela" en la página 368
Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza un clúster adicional para aislar los mensajes a una cola de clúster determinada.

Establecimiento de la comunicación en un clúster

Es necesario un iniciador de canal para iniciar un canal de comunicación cuando hay un mensaje para entregar. Un escucha de canal espera a iniciar el otro extremo de un canal para recibir el mensaje.

Antes de empezar

Para establecer la comunicación entre los gestores de colas de un clúster, configure un enlace utilizando uno de los protocolos de comunicación soportados. Los protocolos soportados son:

- TCP o LU 6.2 en cualquier plataforma
- Windows NetBIOS o SPX en sistemas Windows

Como parte de esta configuración, también necesita iniciadores de canal y escuchas de canal tal como lo hace con la gestión de colas distribuidas.

Acerca de esta tarea

Todos los gestores de colas de clúster necesitan un iniciador de canal para supervisar la cola de inicio definida por el sistema SYSTEM.CHANNEL.INITQ. SYSTEM.CHANNEL.INITQ es la cola de inicio para todas las colas de transmisión, incluida la cola de transmisión de clúster.

Cada gestor de colas debe tener un escucha de canal. Un programa de escucha de canal espera solicitudes de red entrantes e inicia el canal receptor adecuado cuando es necesario. La implementación de los escuchas de canal es específica de la plataforma, pero hay algunas características comunes.

En todas las plataformas IBM MQ, el escucha se puede iniciar con el mandato **START LISTENER**.

Multi En Multiplatforms, puede iniciar el escucha automáticamente al mismo tiempo que el gestor de colas. Para iniciar el escucha automáticamente, establezca el atributo CONTROL del objeto LISTENER en QMGR o STARTONLY.

z/OS Debe utilizarse un puerto de escucha no compartido (INDISP (QMGR)) para canales CLUSRCVR en z/OS y para canales CLUSSDR en z/OS.

Procedimiento

1. Inicie el iniciador de canal.
   - z/OS En z/OS, hay un iniciador de canal para cada gestor de colas y se ejecuta como un espacio de direcciones separado. Puede iniciarlo mediante el mandato **MQSC** START CHINIT, que se emite como parte del inicio del gestor de colas.
   - ALW En AIX, Linux, and Windows, cuando inicie un gestor de colas, si el atributo del gestor de colas SCHINIT está establecido en QMGR, se inicia un iniciador de canal automáticamente. De lo contrario, puede iniciarse mediante el mandato **runmqsc** START CHINIT o el mandato de control **runmqchi**.
   - IBM i En IBM i, cuando inicie un gestor de colas, si el atributo del gestor de colas SCHINIT está establecido en QMGR, se inicia un iniciador de canal automáticamente. De lo contrario, puede iniciarse mediante el mandato **runmqsc** START CHINIT o el mandato de control **runmqchi**.
2. Inicie el escucha de canal.
   - z/OS En z/OS, utilice el programa de escucha de canal proporcionado por IBM MQ. Para iniciar un escucha de canal de IBM MQ, utilice el mandato **MQSC** START LISTENER, que se emite como parte del inicio del iniciador de canal. Por ejemplo:

     ```
     START LISTENER PORT(1414) TRPTYPE(TCP)
     ```

     o:

     ```
     START LISTENER LUNAME(LONDON.LUNAME) TRPTYPE(LU62)
     ```

Los miembros de un grupo de compartición de colas pueden utilizar un escucha compartido en lugar de un escucha para cada gestor de colas. No utilice escuchas compartidos con clústeres. En concreto, no haga que el parámetro CONNAME del canal CLUSRCVR sea la dirección del escucha compartido del grupo de compartición de colas. Si lo hace, los gestores de colas pueden recibir mensajes para colas para las que no tienen una definición.

- IBM i En IBM i, utilice el programa de escucha de canal proporcionado por IBM MQ. Para iniciar un escucha de canal de IBM MQ, utilice el mandato **CL** STRMQMLSR. Por ejemplo:

```
STRMQMLSR MQMNAME(QM1) PORT(1414)
```

- Windows En Windows, utilice el programa de escucha de canal proporcionado por IBM MQ, o los recursos proporcionados por el sistema operativo.

  Para iniciar el escucha de canal de IBM MQ, utilice el mandato RUNMQLSR. Por ejemplo:

```
RUNMQLSR -t tcp -p 1414 -m QM1
```

- Linux AIX En AIX and Linux, utilice el programa de escucha de canal proporcionado por IBM MQ, o los recursos proporcionados por el sistema operativo; por ejemplo, **inetd** para comunicaciones TCP.

  Para iniciar el escucha de canal de IBM MQ, utilice el mandato **runmqlsr**. Por ejemplo:

```
runmqlsr -t tcp -p 1414 -m QM1
```

  Para utilizar **inetd** para iniciar canales, configure dos archivos:

  a. Edite el archivo /etc/services. Debe estar conectado como superusuario o root. Si la línea siguiente no se encuentra en el archivo, añádala como se muestra a continuación:

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

     donde 1414 es el número de puerto requerido por IBM MQ. Puede cambiar el número de puerto, pero debe coincidir con el número de puerto especificado en el extremo emisor.

  b. Edite el archivo /etc/inetd.conf. Si no tiene la línea siguiente en ese archivo, añádala como se muestra a continuación:

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

     donde *MQ_INSTALLATION_PATH* se sustituye por el directorio de alto nivel en el que está instalado IBM MQ.

  Las actualizaciones se activan después de que **inetd** ha vuelto a leer los archivos de configuración. Emita los siguientes mandatos desde el ID de usuario root:

  AIX En AIX:

```
refresh -s inetd
```

  Linux En Linux:

  a. Busque el ID de proceso de **inetd** con el mandato:

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

b. Ejecute el mandato apropiado.

Para Linux:

```
kill -1 inetd processid
```

Configurar un nuevo clúster

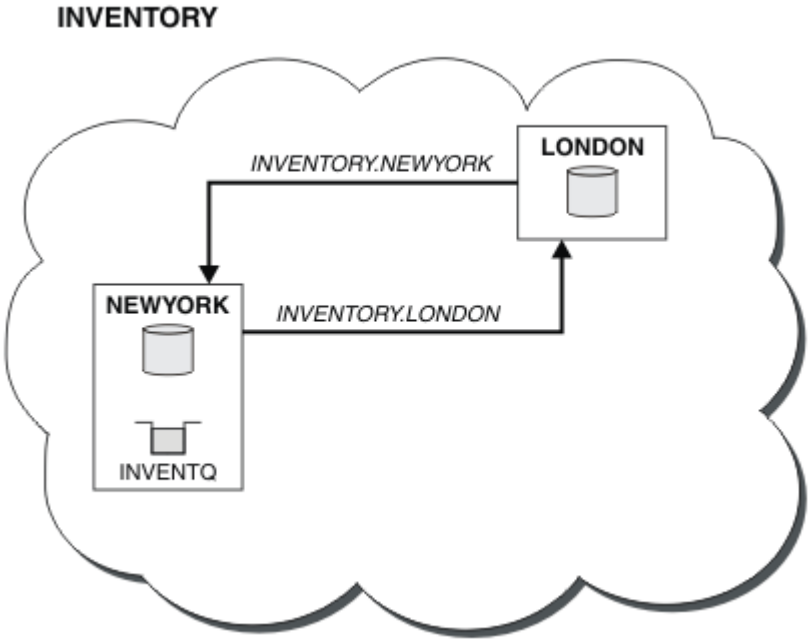
Siga estas instrucciones para configurar el clúster de ejemplo. Instrucciones separadas describen la configuración del clúster en TCP/IP, LU 6.2 y con una única cola de transmisión o varias colas de transmisión. Pruebe los trabajos del clúster enviando un mensaje de un gestor de colas a otro.

Antes de empezar

- En vez de seguir estas instrucciones, puede utilizar uno de los asistentes proporcionados con IBM MQ Explorer para crear un clúster como el que ha creado esta tarea. Pulse con el botón secundario del ratón en la carpeta Clústeres de gestores de colas y, a continuación, pulse **Nuevo** > **Clúster de gestores de colas** y siga las instrucciones proporcionadas en el asistente.
- Para obtener información de fondo que le ayude a comprender los pasos que se siguen para configurar un clúster, consulte "Definición de cola de clúster" en la página 310, Canales de clúster y Escuchas.

Acerca de esta tarea

Está configurando una nueva red de IBM MQ para una cadena de tiendas. La cadena tiene dos sucursales, una en Londres y la otra en Nueva York. Los datos y las aplicaciones para cada sucursal se incluyen en sistemas que ejecutan gestores de colas separados. Los dos gestores de colas se llaman LONDON y NEWYORK. La aplicación de inventario se ejecuta en el sistema en Nueva York, conectado al gestor de colas NEWYORK. La aplicación se activa por la llegada de mensajes en la cola INVENTQ, alojada por NEWYORK. Los dos gestores de colas, LONDON y NEWYORK, se deben enlazar en un clúster llamado INVENTORY, de forma que ambos pueden colocar mensajes en INVENTQ.



Este es el aspecto de este clúster:

Puede configurar cada gestor de colas del clúster para que envíe mensajes a otros gestores de colas del clúster mediante distintas colas de transmisión de clúster.

Las instrucciones para configurar el clúster varían un poco según el protocolo de transporte, el número de colas de transmisión o la plataforma. Puede elegir entre tres combinaciones. El procedimiento de verificación permanece igual para todas las combinaciones.

INVENTORY es un clúster pequeño. Sin embargo, es útil como prueba de concepto. Lo que es importante comprender sobre este clúster es el ámbito que ofrece para futuras mejoras.

Procedimiento

- "Configuración de un clúster utilizando TCP/IP con una sola cola de transmisión por gestor de colas" en la página 325
- "Configuración de un clúster en TCP/IP utilizando múltiples colas de transmisión por gestor de colas" en la página 328
- z/OS "Configurar un clúster utilizando LU 6.2 en z/OS" en la página 331
- "Verificación del clúster" en la página 333

Conceptos relacionados

Clústeres

Comparación de agrupación en clúster y gestión de colas distribuidas

Componentes de un clúster

Tareas relacionadas

"Configuración de un clúster de gestores de colas" en la página 309

Los clústeres proporcionan un mecanismo para interconectar gestores de colas de forma que simplifica la configuración inicial y la gestión continua. Puede definir componentes de clúster, y crear y gestionar los clústeres.

Configuración de un clúster utilizando TCP/IP con una sola cola de transmisión por gestor de colas

Este es uno de los tres temas que describen configuraciones diferentes para un clúster simple.

Antes de empezar

Para obtener una visión general del clúster que se está creando, consulte "Configurar un nuevo clúster" en la página 324.

El atributo de gestor de colas, **DEFCLXQ**, debe permanecer con el valor predeterminado, SCTQ.

Acerca de esta tarea

Siga estos pasos para configurar un clúster en Multiplatforms utilizando el protocolo de transporte TCP/IP.

z/OS En z/OS, debe seguir las instrucciones de "Defining a TCP connection on z/OS" en la página 996 para configurar la conexión TCP/IP, en lugar de definir los escuchas en el paso "4" en la página 326. De lo contrario, los pasos son los mismos para z/OS, pero los mensajes de error se escriben en la consola, en lugar de en el registro de errores del gestor de colas.

Procedimiento

1. Decida sobre la organización del clúster y su nombre.

   Ha decidido enlazar los dos gestores de colas, LONDON y NEWYORK, en un clúster. Un clúster con sólo dos gestores de colas ofrece sólo un beneficio marginal respecto a una red que va a utilizar colas distribuidas. Es una buena manera de empezar y proporciona un ámbito para una futura expansión. Cuando abra nuevas sucursales de la tienda, podrá añadir los nuevos gestores de colas en el clúster fácilmente. Añadir nuevos gestores de colas no interrumpe la red existente; consulte "Añadir un gestor de colas a un clúster" en la página 335.

Por el momento, la única aplicación que está ejecutando es la aplicación de inventario. El nombre del clúster es INVENTORY.

2. Decida qué gestores de colas van a contener repositorios completos.

   En cualquier clúster que deba designar, como mínimo, un gestor de colas, o preferiblemente dos, para contener repositorios completos. En este ejemplo, sólo hay dos gestores de colas, LONDON y NEWYORK, ambos contienen repositorios completos.

   a. Puede realizar los pasos restantes en cualquier orden.

   b. A medida que avance a través de los pasos, los mensajes de aviso podrían escribirse en el registro del gestor de colas. Los mensajes son el resultado de definiciones que faltan y que todavía tiene que añadir.

   ```
   Examples of the responses to the commands are shown in a box
   like this after each step in this task.
   These examples show the responses returned by IBM MQ for AIX.
   The responses vary on other platforms.
   ```

   c. Antes de continuar con estos pasos, asegúrese de que los gestores de colas se hayan iniciado.

3. Modifique las definiciones del gestor de colas para añadir definiciones de repositorio.

   En cada gestor de colas que va a contener un repositorio completo, modifique la definición del gestor de colas local, utilizando el mandato ALTER QMGR y especificando el atributo REPOS:

   ```
   ALTER QMGR REPOS(INVENTORY)
   ```

   ```
   1 : ALTER QMGR REPOS(INVENTORY)
   AMQ8005: IBM MQ queue manager changed.
   ```

   Por ejemplo, si especifica:

   a. `runmqsc LONDON`

   b. `ALTER QMGR REPOS(INVENTORY)`

   LONDON se cambia a un repositorio completo.

4. Defina los escuchas.

   Defina un escucha que acepte solicitudes de red de otros gestores de colas para cada gestor de colas del clúster. En el gestor de colas LONDON, emita el mandato siguiente:

   ```
   DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR)
   ```

   El atributo CONTROL garantiza que el escucha se inicie y se detenga cuando lo haga el gestor de colas.

   El escucha no se inicia cuando se define, por lo que se debe iniciar manualmente la primera vez, con el mandato MQSC siguiente:

   ```
   START LISTENER(LONDON_LS)
   ```

   Emita mandatos similares para todos los demás gestores de colas del clúster, cambiando el nombre del escucha para cada uno.

   Hay varias formas de definir estos escuchas, tal como se muestra en Escuchas.

5. Defina el canal CLUSRCVR para el gestor de colas LONDON.

   En cada gestor de colas de un clúster, defina un canal de clúster receptor en el que el gestor de colas pueda recibir mensajes. Consulte Canal de clúster receptor: CLUSRCVR. El canal CLUSRCVR define el nombre de conexión del gestor de colas. El nombre de conexión se almacena en los

repositorios, donde otros gestores de colas pueden consultarlo. La palabra clave CLUSTER muestra la disponibilidad del gestor de colas para recibir mensajes de otros gestores de colas del clúster.

En este ejemplo, el nombre de canal es INVENTORY.LONDON y el nombre de conexión (CONNAME) es la dirección de red de la máquina en la que reside el gestor de colas, que es LONDON.CHSTORE.COM. La dirección de red se puede especificar como un nombre de host DNS alfanumérico, o una dirección IP en formato IPv4, o bien en formato decimal con puntos. Por ejemplo, 192.0.2.0, o el formato hexadecimal IPv6; por ejemplo 2001:DB8:0204:acff:fe97:2c34:fde0:3485. No se especifica el Número de puerto, por lo que se utiliza el puerto predeterminado (1414).

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
AMQ8014: WebSphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

6. Defina el canal CLUSRCVR para el gestor de colas NEWYORK.

   z/OS Si el escucha de canal utiliza el puerto predeterminado, normalmente 1414, y el clúster no incluye un gestor de colas en z/OS, puede omitir CONNAME.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-receiver channel for queue manager NEWYORK')
   ```

7. Defina el canal CLUSSDR en el gestor de colas LONDON.

   El administrador debe definir un canal CLUSSDR desde cada gestor de colas de repositorio completo a cualquier otro gestor de colas de repositorio completo del clúster. Consulte Canal de clúster emisor: CLUSSDR. En este caso, sólo hay dos gestores de colas, ambos contienen repositorios completos. Ambos deben tener un canal CLUSSDR definido manualmente que apunte al canal CLUSRCVR definido en el otro gestor de colas. Los nombres de canal proporcionados en las definiciones CLUSSDR deben coincidir con los nombres de canal en las definiciones CLUSRCVR correspondientes. Cuando un gestor de colas tiene definiciones tanto para el canal de clúster receptor, como para el canal de clúster emisor en el mismo clúster, se inicia el canal de clúster emisor.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
   ```

   ```
   1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
   AMQ8014: WebSphere MQ channel created.
   07/09/98 13:00:18  Channel program started.
   ```

8. Defina el canal CLUSSDR en el gestor de colas NEWYORK.

   ```
   DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from NEWYORK to repository at LONDON')
   ```

9. Defina la cola de clúster INVENTQ

   Defina la cola INVENTQ en el gestor de colas NEWYORK, especificando la palabra clave CLUSTER.

   ```
   DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
   ```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: WebSphere MQ queue created.
```

La palabra clave CLUSTER provoca que se anuncie la cola en el clúster. Tan pronto como la cola se define, pasa a estar a disponible en los otros gestores de colas del clúster. Pueden enviarle mensajes sin tener que marcar una definición de cola remota para ello.

Todas las definiciones se han completado. En todas las plataformas, inicie un programa de escucha en cada gestor de colas. El programa de escucha espera peticiones de red entrantes e inicia el canal de clúster receptor cuando es necesario.

Qué hacer a continuación

Ahora está preparado para verificar el clúster.

Tareas relacionadas

"Configuración de un clúster en TCP/IP utilizando múltiples colas de transmisión por gestor de colas" en la página 328

Este es uno de los tres temas que describen configuraciones diferentes para un clúster simple.

"Configurar un clúster utilizando LU 6.2 en z/OS" en la página 331

Este es uno de los temas de árbol que describen diferentes configuraciones para un clúster simple.

Configuración de un clúster en TCP/IP utilizando múltiples colas de transmisión por gestor de colas

Este es uno de los tres temas que describen configuraciones diferentes para un clúster simple.

Antes de empezar

Para obtener una visión general del clúster que se está creando, consulte "Configurar un nuevo clúster" en la página 324.

Acerca de esta tarea

Siga estos pasos para configurar un clúster en Multiplatforms utilizando el protocolo de transporte TCP/IP.Los gestores de colas de repositorio se configuran para utilizar una cola de transmisión de clúster diferente para enviar mensajes entre sí y a los demás gestores de colas del clúster. Si añade gestores de colas al clúster que también van a utilizar diferentes colas de transmisión, siga la tarea, "Añadir un gestor de colas a un clúster: colas de transmisión separadas" en la página 337.

Procedimiento

1. Decida sobre la organización del clúster y su nombre.

   Ha decidido enlazar los dos gestores de colas, LONDON y NEWYORK, en un clúster. Un clúster con sólo dos gestores de colas ofrece sólo un beneficio marginal respecto a una red que va a utilizar colas distribuidas. Es una buena manera de empezar y proporciona un ámbito para una futura expansión. Cuando abra nuevas sucursales de la tienda, podrá añadir los nuevos gestores de colas en el clúster fácilmente. Añadir nuevos gestores de colas no interrumpe la red existente; consulte "Añadir un gestor de colas a un clúster" en la página 335.

   Por el momento, la única aplicación que está ejecutando es la aplicación de inventario. El nombre del clúster es INVENTORY.

2. Decida qué gestores de colas van a contener repositorios completos.

   En cualquier clúster que deba designar, como mínimo, un gestor de colas, o preferiblemente dos, para contener repositorios completos. En este ejemplo, sólo hay dos gestores de colas, LONDON y NEWYORK, ambos contienen repositorios completos.

   a. Puede realizar los pasos restantes en cualquier orden.

b. A medida que avance a través de los pasos, los mensajes de aviso podrían escribirse en el registro del gestor de colas. Los mensajes son el resultado de definiciones que faltan y que todavía tiene que añadir.

```
Examples of the responses to the commands are shown in a box
like this after each step in this task.
These examples show the responses returned by IBM MQ for AIX.
The responses vary on other platforms.
```

c. Antes de continuar con estos pasos, asegúrese de que los gestores de colas se hayan iniciado.

3. Modifique las definiciones del gestor de colas para añadir definiciones de repositorio.

   En cada gestor de colas que va a contener un repositorio completo, modifique la definición del gestor de colas local, utilizando el mandato ALTER QMGR y especificando el atributo REPOS:

```
ALTER QMGR REPOS(INVENTORY)
```

```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: IBM MQ queue manager changed.
```

   Por ejemplo, si especifica:

   a. `runmqsc LONDON`

   b. `ALTER QMGR REPOS(INVENTORY)`

   LONDON se cambia a un repositorio completo.

4. Modifique las definiciones de gestor de colas para crear colas de transmisión de clúster separadas para cada destino.

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

   En cada gestor de colas que añada al clúster, decida si va a utilizar colas de transmisión distintas o no. Consulte los temas “Añadir un gestor de colas a un clúster” en la página 335 y “Añadir un gestor de colas a un clúster: colas de transmisión separadas” en la página 337.

5. Defina los escuchas.

   Defina un escucha que acepte solicitudes de red de otros gestores de colas para cada gestor de colas del clúster. En el gestor de colas LONDON, emita el mandato siguiente:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR)
```

   El atributo CONTROL garantiza que el escucha se inicie y se detenga cuando lo haga el gestor de colas.

   El escucha no se inicia cuando se define, por lo que se debe iniciar manualmente la primera vez, con el mandato MQSC siguiente:

```
START LISTENER(LONDON_LS)
```

   Emita mandatos similares para todos los demás gestores de colas del clúster, cambiando el nombre del escucha para cada uno.

   Hay varias formas de definir estos escuchas, tal como se muestra en Escuchas.

6. Defina el canal CLUSRCVR para el gestor de colas LONDON.

   En cada gestor de colas de un clúster, defina un canal de clúster receptor en el que el gestor de colas pueda recibir mensajes. Consulte Canal de clúster receptor: CLUSRCVR. El canal CLUSRCVR define el nombre de conexión del gestor de colas. El nombre de conexión se almacena en los

repositorios, donde otros gestores de colas pueden consultarlo. La palabra clave CLUSTER muestra la disponibilidad del gestor de colas para recibir mensajes de otros gestores de colas del clúster.

En este ejemplo, el nombre de canal es INVENTORY.LONDON y el nombre de conexión (CONNAME) es la dirección de red de la máquina en la que reside el gestor de colas, que es LONDON.CHSTORE.COM. La dirección de red se puede especificar como un nombre de host DNS alfanumérico, o una dirección IP en formato IPv4, o bien en formato decimal con puntos. Por ejemplo, 192.0.2.0, o el formato hexadecimal IPv6; por ejemplo 2001:DB8:0204:acff:fe97:2c34:fde0:3485. No se especifica el Número de puerto, por lo que se utiliza el puerto predeterminado (1414).

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
AMQ8014: WebSphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

7. Defina el canal CLUSRCVR para el gestor de colas NEWYORK.

   z/OS Si el escucha de canal utiliza el puerto predeterminado, normalmente 1414, y el clúster no incluye un gestor de colas en z/OS, puede omitir CONNAME.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-receiver channel for queue manager NEWYORK')
   ```

8. Defina el canal CLUSSDR en el gestor de colas LONDON.

   El administrador debe definir un canal CLUSSDR desde cada gestor de colas de repositorio completo a cualquier otro gestor de colas de repositorio completo del clúster. Consulte Canal de clúster emisor: CLUSSDR. En este caso, sólo hay dos gestores de colas, ambos contienen repositorios completos. Ambos deben tener un canal CLUSSDR definido manualmente que apunte al canal CLUSRCVR definido en el otro gestor de colas. Los nombres de canal proporcionados en las definiciones CLUSSDR deben coincidir con los nombres de canal en las definiciones CLUSRCVR correspondientes. Cuando un gestor de colas tiene definiciones tanto para el canal de clúster receptor, como para el canal de clúster emisor en el mismo clúster, se inicia el canal de clúster emisor.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
   ```

   ```
   1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
   AMQ8014: WebSphere MQ channel created.
   07/09/98 13:00:18  Channel program started.
   ```

9. Defina el canal CLUSSDR en el gestor de colas NEWYORK.

   ```
   DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from NEWYORK to repository at LONDON')
   ```

10. Defina la cola de clúster INVENTQ

    Defina la cola INVENTQ en el gestor de colas NEWYORK, especificando la palabra clave CLUSTER.

    ```
    DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
    ```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: WebSphere MQ queue created.
```

La palabra clave CLUSTER provoca que se anuncie la cola en el clúster. Tan pronto como la cola se define, pasa a estar a disponible en los otros gestores de colas del clúster. Pueden enviarle mensajes sin tener que marcar una definición de cola remota para ello.

Todas las definiciones se han completado. En todas las plataformas, inicie un programa de escucha en cada gestor de colas. El programa de escucha espera peticiones de red entrantes e inicia el canal de clúster receptor cuando es necesario.

Qué hacer a continuación

Ahora está preparado para verificar el clúster.

Tareas relacionadas

"Configuración de un clúster utilizando TCP/IP con una sola cola de transmisión por gestor de colas" en la página 325

Este es uno de los tres temas que describen configuraciones diferentes para un clúster simple.

"Configurar un clúster utilizando LU 6.2 en z/OS" en la página 331

Este es uno de los temas de árbol que describen diferentes configuraciones para un clúster simple.

z/OS *Configurar un clúster utilizando LU 6.2 en z/OS*

Este es uno de los temas de árbol que describen diferentes configuraciones para un clúster simple.

Antes de empezar

Para obtener una visión general del clúster que se está creando, consulte "Configurar un nuevo clúster" en la página 324.

Procedimiento

1. Decida sobre la organización del clúster y su nombre.

   Ha decidido enlazar los dos gestores de colas, LONDON y NEWYORK, en un clúster. Un clúster con sólo dos gestores de colas ofrece sólo un beneficio marginal respecto a una red que va a utilizar colas distribuidas. Es una buena manera de empezar y proporciona un ámbito para una futura expansión. Cuando abra nuevas sucursales de la tienda, podrá añadir los nuevos gestores de colas en el clúster fácilmente. Añadir nuevos gestores de colas no interrumpe la red existente; consulte "Añadir un gestor de colas a un clúster" en la página 335.

   Por el momento, la única aplicación que está ejecutando es la aplicación de inventario. El nombre del clúster es INVENTORY.
2. Decida qué gestores de colas van a contener repositorios completos.

   En cualquier clúster que deba designar, como mínimo, un gestor de colas, o preferiblemente dos, para contener repositorios completos. En este ejemplo, sólo hay dos gestores de colas, LONDON y NEWYORK, ambos contienen repositorios completos.

   a. Puede realizar los pasos restantes en cualquier orden.
   b. A medida que avance por los pasos, los mensajes de aviso podrían escribirse en la consola del sistema de z/OS. Los mensajes son el resultado de definiciones que faltan y que todavía tiene que añadir.
   c. Antes de continuar con estos pasos, asegúrese de que los gestores de colas se hayan iniciado.
3. Modifique las definiciones del gestor de colas para añadir definiciones de repositorio.

En cada gestor de colas que va a contener un repositorio completo, modifique la definición del gestor de colas local, utilizando el mandato ALTER QMGR y especificando el atributo REPOS:

```
ALTER QMGR REPOS(INVENTORY)
```

```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: IBM MQ queue manager changed.
```

Por ejemplo, si especifica:

a. `runmqsc LONDON`

b. `ALTER QMGR REPOS(INVENTORY)`

LONDON se cambia a un repositorio completo.

4. Defina los escuchas.

z/OS Consulte El iniciador de canal en z/OS y “Receiving on LU 6.2” en la página 1000.

El escucha no se inicia cuando se define, por lo que se debe iniciar manualmente la primera vez, con el mandato MQSC siguiente:

```
START LISTENER(LONDON_LS)
```

Emita mandatos similares para todos los demás gestores de colas del clúster, cambiando el nombre del escucha para cada uno.

5. Defina el canal CLUSRCVR para el gestor de colas LONDON.

En cada gestor de colas de un clúster, defina un canal de clúster receptor en el que el gestor de colas pueda recibir mensajes. Consulte Canal de clúster receptor: CLUSRCVR. El canal CLUSRCVR define el nombre de conexión del gestor de colas. El nombre de conexión se almacena en los repositorios, donde otros gestores de colas pueden consultarlo. La palabra clave CLUSTER muestra la disponibilidad del gestor de colas para recibir mensajes de otros gestores de colas del clúster.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager LONDON')
AMQ8014: WebSphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

6. Defina el canal CLUSRCVR para el gestor de colas NEWYORK.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(NEWYORK.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager NEWYORK')
```

7. Defina el canal CLUSSDR en el gestor de colas LONDON.

El administrador debe definir un canal CLUSSDR desde cada gestor de colas de repositorio completo a cualquier otro gestor de colas de repositorio completo del clúster. Consulte Canal de clúster emisor: CLUSSDR. En este caso, sólo hay dos gestores de colas, ambos contienen repositorios completos. Ambos deben tener un canal CLUSSDR definido manualmente que apunte al canal CLUSRCVR definido en el otro gestor de colas. Los nombres de canal proporcionados en las definiciones CLUSSDR deben coincidir con los nombres de canal en las definiciones CLUSRCVR correspondientes. Cuando un gestor

de colas tiene definiciones tanto para el canal de clúster receptor, como para el canal de clúster emisor en el mismo clúster, se inicia el canal de clúster emisor.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(CPIC) CLUSTER(INVENTORY)
DESCR('LU62 Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(NEWYORK.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: WebSphere MQ channel created.
07/09/98 13:00:18  Channel program started.
```

8. Defina el canal CLUSSDR en el gestor de colas NEWYORK.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
DESCR('LU62 Cluster-sender channel from NEWYORK to repository at LONDON')
```

9. Defina la cola de clúster INVENTQ

   Defina la cola INVENTQ en el gestor de colas NEWYORK, especificando la palabra clave CLUSTER.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: WebSphere MQ queue created.
```

   La palabra clave CLUSTER provoca que se anuncie la cola en el clúster. Tan pronto como la cola se define, pasa a estar a disponible en los otros gestores de colas del clúster. Pueden enviarle mensajes sin tener que marcar una definición de cola remota para ello.

   Todas las definiciones se han completado. En todas las plataformas, inicie un programa de escucha en cada gestor de colas. El programa de escucha espera peticiones de red entrantes e inicia el canal de clúster receptor cuando es necesario.

Qué hacer a continuación

Ahora está preparado para verificar el clúster.

Tareas relacionadas

“Configuración de un clúster utilizando TCP/IP con una sola cola de transmisión por gestor de colas” en la página 325

Este es uno de los tres temas que describen configuraciones diferentes para un clúster simple.

“Configuración de un clúster en TCP/IP utilizando múltiples colas de transmisión por gestor de colas” en la página 328

Este es uno de los tres temas que describen configuraciones diferentes para un clúster simple.

Verificación del clúster

Los temas de igual describen tres configuraciones diferentes para un clúster simple. En este tema se explica cómo verificar el clúster.

Antes de empezar

En este tema se presupone que está verificando un clúster que ha creado a través de una de las tareas siguientes:

- “Configuración de un clúster utilizando TCP/IP con una sola cola de transmisión por gestor de colas” en la página 325.

- “Configuración de un clúster en TCP/IP utilizando múltiples colas de transmisión por gestor de colas” en la página 328.
- z/OS “Configurar un clúster utilizando LU 6.2 en z/OS” en la página 331.

Para obtener una visión general del clúster que se ha creado, consulte “Configurar un nuevo clúster” en la página 324.

Acerca de esta tarea

Puede verificar el clúster de una o varias de las formas siguientes:

1. Ejecutando mandatos administrativos para visualizar atributos de clúster y canal.
2. Ejecute los programas de ejemplo para enviar y recibir mensajes en una cola de clúster.
3. Escriba sus propios programas para enviar un mensaje de solicitud a una cola de clúster y responder con mensajes de respuesta a una cola de respuesta sin clúster.

Procedimiento

Emita mandatos DISPLAY **runmqsc** para verificar el clúster que ha configurado.

Las respuestas que verá deberían ser como las respuestas de los pasos que aparecen a continuación.

1. Desde el gestor de colas NEWYORK, ejecute el mandato **DISPLAY CLUSQMGR**:

```
dis clusqmgr(*)
```

```
1 : dis clusqmgr(*)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(NEWYORK)       CLUSTER(INVENTORY)
CHANNEL(INVENTORY.NEWYORK)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(LONDON)       CLUSTER(INVENTORY)
CHANNEL(INVENTORY.LONDON)
```

2. Desde el gestor de colas NEWYORK, ejecute el mandato **DISPLAY CHANNEL STATUS**:

```
dis chstatus(*)
```

```
1 : dis chstatus(*)
AMQ8417: Display Channel Status details.
CHANNEL(INVENTORY.NEWYORK)  XMITQ( )
CONNAME(192.0.2.0)      CURRENT
CHLTYPE(CLUSRCVR)       STATUS(RUNNING)
RQMNAME(LONDON)
AMQ8417: Display Channel Status details.
CHANNEL(INVENTORY.LONDON) XMITQ(SYSTEM.CLUSTER.TRANSMIT.INVENTORY.LONDON)
CONNAME(192.0.2.1)      CURRENT
CHLTYPE(CLUSSDR)       STATUS(RUNNING)
RQMNAME(LONDON)
```

Envíe mensajes entre los dos gestores de colas, utilizando **amqsput**.

3. En LONDON , ejecute el mandato **amqsput INVENTQ LONDON**.

   Escriba algunos mensajes, seguidos de una línea en blanco.
4. En NEWYORK , ejecute el mandato **amqsget INVENTQ NEWYORK**.

   Ahora verá los mensajes que ha especificado en LONDON. Transcurridos 15 segundos, el programa finaliza.

Envíe mensajes entre los dos gestores de colas, utilizando sus propios programas.

En los pasos siguientes, LONDON coloca un mensaje en INVENTQ en NEWYORK y recibe una respuesta en su cola LONDON_reply.

5. En LONDON transfiera un mensaje a la cola.
   a) Defina una cola local denominada LONDON_reply.
   b) Establezca las opciones MQOPEN en MQOO_OUTPUT.
   c) Emita la llamada MQOPEN para abrir la cola INVENTQ.
   d) Establezca el nombre *ReplyToQ* en el descriptor de mensaje en LONDON_reply.
   e) Emita la llamada MQPUT para colocar el mensaje.
   f) Confirme el mensaje.
6. En NEWYORK reciba el mensaje en la cola de clúster y transfiera una respuesta a la cola de respuesta.
   a) Establezca las opciones MQOPEN en MQOO_BROWSE.
   b) Emita la llamada MQOPEN para abrir la cola INVENTQ.
   c) Emita la llamada MQGET para obtener el mensaje de INVENTQ.
   d) Recupere el nombre *ReplyToQ* del descriptor de mensaje.
   e) Coloque el nombre *ReplyToQ* en el campo ObjectName del descriptor de objeto.
   f) Establezca las opciones MQOPEN en MQOO_OUTPUT.
   g) Emita la llamada MQOPEN para abrir LONDON_reply en el gestor de colas LONDON.
   h) Emita la llamada MQPUT para colocar el mensaje en LONDON_reply.
7. En LONDON reciba la respuesta.
   a) Establezca las opciones MQOPEN en MQOO_BROWSE.
   b) Emita la llamada MQOPEN para abrir la cola LONDON_reply.
   c) Emita la llamada MQGET para obtener el mensaje de LONDON_reply.

Añadir un gestor de colas a un clúster

Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando la cola de transmisión de clúster única SYSTEM.CLUSTER.TRANSMIT.QUEUE.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- El clúster INVENTORY se ha configurado tal como se describe en “Configurar un nuevo clúster” en la página 324. Contiene dos gestores de colas, LONDON y NEWYORK, que contienen depósitos completos.
- El gestor de colas PARIS es propiedad de la instalación primaria. De lo contrario, debe ejecutar el mandato **setmqenv** para configurar el entorno de mandato para la instalación a la que pertenece PARIS.
- La conectividad TCP existe entre los tres sistemas, y el gestor de colas está configurado con un escucha TCP que se inicia bajo el control del gestor de colas.

Acerca de esta tarea

1. Se está abriendo una nueva sucursal de la cadena de tiendas en París y desea añadir un gestor de colas llamado PARIS al clúster.
2. El gestor de colas PARIS envía actualizaciones de inventario a la aplicación que se ejecuta en el sistema de Nueva York, colocando mensajes en la cola INVENTQ.

Siga estos pasos para añadir un gestor de colas a un clúster.

Procedimiento

1. Decida a qué repositorio completo hace referencia primero PARIS.

   Cada gestor de colas de un clúster debe hacer referencia a uno de los dos repositorios completos. El gestor de colas recopila información sobre el clúster de un repositorio completo y así crea su propio repositorio parcial. Elija cualquiera de los dos repositorios como el repositorio completo. En cuanto se añade un nuevo gestor de colas al clúster, se informa inmediatamente sobre el otro repositorio. La información sobre los cambios en un gestor de colas se envía directamente a dos repositorios. En este ejemplo, enlaza PARIS al gestor de colas LONDON, sólo por razones geográficas.

   Nota: Realice los pasos restantes en cualquier orden, después de que se inicie el gestor de colas PARIS.
2. Defina un canal CLUSRCVR en el gestor de colas PARIS.

   Cada gestor de colas de un clúster debe definir un canal de clúster receptor en el que puede recibir mensaje. En PARIS, defina:

   ```
   DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-receiver channel for queue manager PARIS')
   ```

   El canal de clúster receptor anuncia la disponibilidad del gestor de colas para recibir mensajes de otros gestores de colas en el clúster INVENTORY. No cree definiciones en otros gestores de colas para un extremo emisor al canal de clúster receptor INVENTORY.PARIS. Otras definiciones se crearán automáticamente cuando sea necesario. Consulte Canales de clúster.
3. z/OS

   Inicie el iniciador de canal en IBM MQ for z/OS.
4. Defina un canal CLUSSDR en el gestor de colas PARIS.

   Cuando añade a un gestor de colas a un clúster que no es un repositorio completo, simplemente define un canal de clúster emisor para crear una conexión inicial con un repositorio completo. Consulte Canal de clúster emisor: CLUSSDR.

   En PARIS, cree la siguiente definición para un canal CLUSSDR llamado INVENTORY.LONDON para el gestor de colas con la dirección de red LONDON.CHSTORE.COM.

   ```
   DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from PARIS to repository at LONDON')
   ```
5. Opcional: Si está añadiendo a un clúster un gestor de colas que anteriormente se había eliminado del mismo clúster, compruebe que se muestre ahora como un miembro del clúster. Si no es así, realice los siguientes pasos adicionales:
   a) Emita el mandato **REFRESH CLUSTER** en el gestor de colas que está añadiendo.

      Este paso es necesario detiene los canales del clúster y entrega a la memoria caché de clúster local un nuevo conjunto de números de secuencia que con toda seguridad están al día dentro del resto del clúster.

      ```
      REFRESH CLUSTER(INVENTORY) REPOS(YES)
      ```

      Nota: Para los clústeres de gran tamaño, el uso del mandato **REFRESH CLUSTER** puede interrumpir el clúster mientras está en curso y, a partir de entonces, de nuevo a intervalos de 27 días cuando los objetos de clúster envían automáticamente actualizaciones de estado a todos los gestores de colas interesados. Consulte La renovación en un clúster grande puede afectar el rendimiento y la disponibilidad del clúster.
   b) Reinicie el canal CLUSSDR.

      Por ejemplo mediante el mandato START CHANNEL.

c) Reinicie el canal CLUSRCVR.

Resultados

La figura siguiente muestra el clúster configurado por esta tarea.

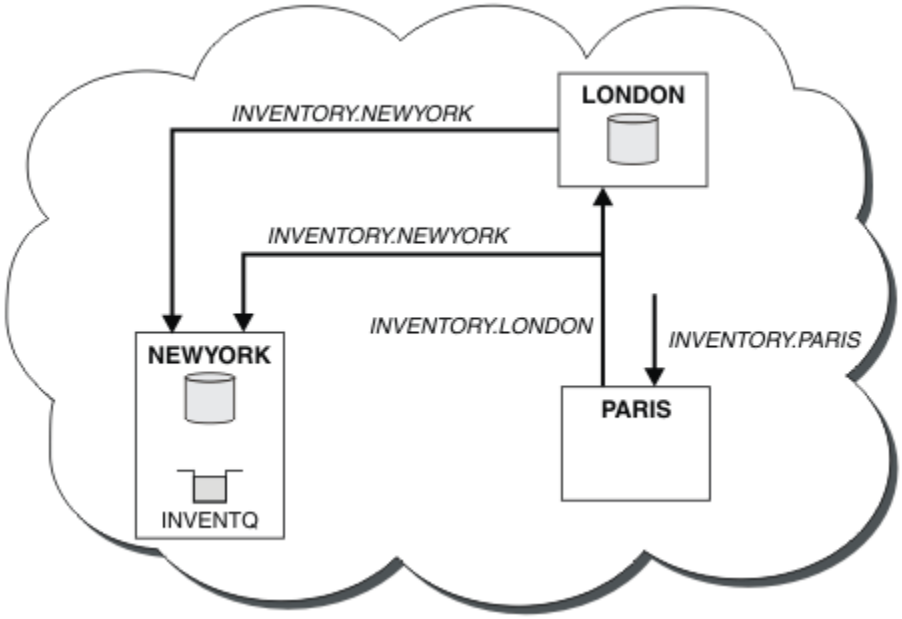


Figura 39. El clúster INVENTORY con tres gestores de colas

Haciendo sólo dos definiciones, una definición CLUSRCVR y una definición CLUSSDR, añadimos el gestor de colas PARIS al clúster.

Ahora, el gestor de colas PARIS se informa, en el repositorio completo en LONDON, que la cola INVENTQ está alojada por el gestor de colas NEWYORK. Cuando una aplicación alojada por el sistema en París intenta colocar mensajes en la cola INVENTQ, PARIS define automáticamente un canal de clúster emisor para conectarse al canal de clúster receptor INVENTORY.NEWYORK. La aplicación puede recibir respuestas cuando su nombre de gestor de colas se especifica como el gestor de colas de destino y se proporciona una cola de respuesta.

Añadir un gestor de colas a un clúster: colas de transmisión separadas

Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando varias colas de transmisión de clúster.

Antes de empezar

- El gestor de colas no es miembro de ningún clúster.
- El clúster existe; hay un depósito completo al que este gestor de colas puede conectarse directamente y el depósito está disponible. Para ver los pasos para crear el clúster, consulte "Configurar un nuevo clúster" en la página 324.

Acerca de esta tarea

Esta tarea es una alternativa a "Añadir un gestor de colas a un clúster" en la página 335, donde puede añadir un gestor de colas a un clúster que coloca mensajes de clúster en una cola de transmisión única.

En esta tarea, añade un gestor de colas a un clúster que crea automáticamente colas de transmisión de clúster diferentes para cada canal de clúster emisor.

Para mantener el número de definiciones de colas pequeñas, el valor predeterminado es utilizar una sola cola de transmisión. Utilizar distintas colas de transmisión resulta beneficioso si desea supervisar

el tráfico destinado a diferentes gestores de colas y distintos clústeres. Es posible que también quiera separar el tráfico a distintos destinos para conseguir los objetivos de aislamiento o de rendimiento.

Procedimiento

1. Modifique el tipo de cola de transmisión del canal de clúster predeterminado.

   Modifique el gestor de colas PARIS:

   ```
   ALTER QMGR DEFCLXQ(CHANNEL)
   ```

   Cada vez que el gestor de colas crea un canal de clúster emisor para enviar un mensaje a un gestor de colas, éste crea una cola de transmisión de clúster. La cola de transmisión solamente es utilizada por este canal de clúster emisor. La cola de transmisión es dinámica permanente. Se crea a partir de la cola modelo, SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE, con el nombre SYSTEM.CLUSTER.TRANSMIT.*ChannelName*.

   Atención: Si utiliza SYSTEM.CLUSTER.TRANSMIT.QUEUES dedicado con un gestor de colas que se ha actualizado desde una versión del producto anterior a IBM WebSphere MQ 7.5, asegúrese de que SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE tiene la opción SHARE/NOSHARE establecida en **SHARE**.

2. Decida a qué repositorio completo hace referencia primero PARIS.

   Cada gestor de colas de un clúster debe hacer referencia a uno de los dos repositorios completos. El gestor de colas recopila información sobre el clúster de un repositorio completo y así crea su propio repositorio parcial. Elija cualquiera de los dos repositorios como el repositorio completo. En cuanto se añade un nuevo gestor de colas al clúster, se informa inmediatamente sobre el otro repositorio. La información sobre los cambios en un gestor de colas se envía directamente a dos repositorios. En este ejemplo, enlaza PARIS al gestor de colas LONDON, sólo por razones geográficas.

   Nota: Realice los pasos restantes en cualquier orden, después de que se inicie el gestor de colas PARIS.

3. Defina un canal CLUSRCVR en el gestor de colas PARIS.

   Cada gestor de colas de un clúster debe definir un canal de clúster receptor en el que puede recibir mensaje. En PARIS, defina:

   ```
   DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-receiver channel for queue manager PARIS')
   ```

   El canal de clúster receptor anuncia la disponibilidad del gestor de colas para recibir mensajes de otros gestores de colas en el clúster INVENTORY. No cree definiciones en otros gestores de colas para un extremo emisor al canal de clúster receptor INVENTORY.PARIS. Otras definiciones se crearán automáticamente cuando sea necesario. Consulte Canales de clúster.

4. Defina un canal CLUSSDR en el gestor de colas PARIS.

   Cuando añade a un gestor de colas a un clúster que no es un repositorio completo, simplemente define un canal de clúster emisor para crear una conexión inicial con un repositorio completo. Consulte Canal de clúster emisor: CLUSSDR.

   En PARIS, cree la siguiente definición para un canal CLUSSDR llamado INVENTORY.LONDON para el gestor de colas con la dirección de red LONDON.CHSTORE.COM.

   ```
   DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from PARIS to repository at LONDON')
   ```

   El gestor de colas crea automáticamente la cola de transmisión de clúster dinámica permanente SYSTEM.CLUSTER.TRANSMIT.INVENTORY.LONDON a partir de la cola modelo

SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE. Establece el atributo CLCHNAME de la cola de transmisión con el valor INVENTORY.LONDON.

Resultados

La figura siguiente muestra el clúster configurado por esta tarea.

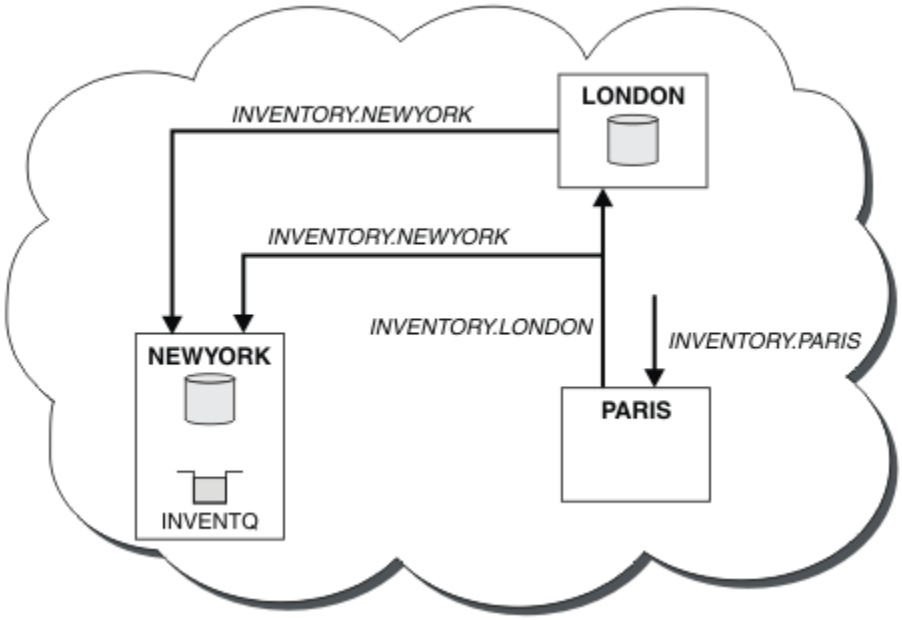


Figura 40. El clúster INVENTORY con tres gestores de colas

Haciendo sólo dos definiciones, una definición CLUSRCVR y una definición CLUSSDR, añadimos el gestor de colas PARIS al clúster.

Ahora, el gestor de colas PARIS se informa, en el repositorio completo en LONDON, que la cola INVENTQ está alojada por el gestor de colas NEWYORK. Cuando una aplicación alojada por el sistema en París intenta colocar mensajes en la cola INVENTQ, PARIS define automáticamente un canal de clúster emisor para conectarse al canal de clúster receptor INVENTORY.NEWYORK. La aplicación puede recibir respuestas cuando su nombre de gestor de colas se especifica como el gestor de colas de destino y se proporciona una cola de respuesta.

Conceptos relacionados

Cómo seleccionar qué tipo de cola de transmisión de clúster se debe utilizar

Tareas relacionadas

Añadir un gestor de colas a un clúster mediante DHCP

Añada un gestor de colas a un clúster, utilizando DHCP. La tarea demuestra la omisión del valor de CONNAME en una definición CLUSRCVR.

Añadir un gestor de colas a un clúster mediante DHCP

Añada un gestor de colas a un clúster, utilizando DHCP. La tarea demuestra la omisión del valor de CONNAME en una definición CLUSRCVR.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

La tarea demuestra dos características especiales:

- La posibilidad de omitir el valor de CONNAME en una definición CLUSRCVR.

- La posibilidad de utilizar +QMNAME+ en una definición CLUSSDR.

z/OS Ninguna de las características se proporciona en z/OS.

Escenario:

- El clúster de INVENTORY se ha configurado como se describe en “Configurar un nuevo clúster” en la página 324. Contiene dos gestores de colas, LONDON y NEWYORK, que contienen depósitos completos.
- Se está abriendo una nueva sucursal de la cadena de tiendas en París y desea añadir un gestor de colas llamado PARIS al clúster.
- El gestor de colas PARIS envía actualizaciones de inventario a la aplicación que se ejecuta en el sistema de Nueva York, colocando mensajes en la cola INVENTQ.
- Existe conectividad de red entre los tres sistemas.
- El protocolo de red es TCP.
- El sistema del gestor de colas PARIS utiliza DHCP, lo que significa que las direcciones IP pueden cambiar al reiniciar el sistema.
- A los canales entre los sistemas PARIS y LONDON se les asigna un nombre siguiendo un convenio de denominación definido. El convenio utiliza el nombre de gestor de colas del gestor de colas de depósito completo en LONDON.
- Los administradores del gestor de colas PARIS no tienen información sobre el nombre del gestor de colas en el depósito de LONDON. El nombre del gestor de colas en el depósito de LONDON está sujeto a cambios.

Acerca de esta tarea

Siga estos pasos para añadir un gestor de colas a un clúster utilizando DHCP.

Procedimiento

1. Decida a qué repositorio completo hace referencia primero PARIS.

   Cada gestor de colas de un clúster debe hacer referencia a uno de los dos repositorios completos. El gestor de colas recopila información sobre el clúster de un repositorio completo y así crea su propio repositorio parcial. Elija cualquiera de los dos repositorios como el repositorio completo. En cuanto se añade un nuevo gestor de colas al clúster, se informa inmediatamente sobre el otro repositorio. La información sobre los cambios en un gestor de colas se envía directamente a dos repositorios. En este ejemplo, decidimos enlazar PARIS al gestor de colas LONDON, sólo por razones geográficas.

   Nota: Realice los pasos restantes en cualquier orden, después de que se inicie el gestor de colas PARIS.
2. Defina un canal CLUSRCVR en el gestor de colas PARIS.

   Cada gestor de colas de un clúster tiene que definir un canal de clúster receptor en el que pueda recibir mensajes. En PARIS, defina:

   ```
   DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR)
   TRPTYPE(TCP) CLUSTER(INVENTORY)
   DESCR('Cluster-receiver channel for queue manager PARIS')
   ```

   El canal de clúster receptor anuncia la disponibilidad del gestor de colas para recibir mensajes de otros gestores de colas en el clúster INVENTORY. No es necesario especificar el CONNAME en el canal de clúster receptor. Puede solicitar IBM MQ para averiguar el nombre de conexión del sistema omitiendo CONNAME o bien especificando CONNAME(' '). IBM MQ genera el valor CONNAME utilizando la dirección IP actual del sistema; consulte CONNAME . No es necesario crear definiciones en otros gestores de colas para un extremo emisor al canal de clúster receptor INVENTORY.PARIS. Otras definiciones se crearán automáticamente cuando sea necesario.
3. Defina un canal CLUSSDR en el gestor de colas PARIS.

Cada gestor de colas de un clúster tiene que definir un canal de clúster emisor en el que pueda enviar mensajes a su repositorio completo inicial. En PARIS, cree la siguiente definición para un canal llamado INVENTORY.+QMNAME+ para el gestor de colas cuya dirección de red es LONDON.CHSTORE.COM.

```
DEFINE CHANNEL(INVENTORY.+QMNAME+) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

4. Opcional: Si está añadiendo a un clúster un gestor de colas que anteriormente se había eliminado del mismo clúster, compruebe que se muestre ahora como un miembro del clúster. Si no es así, realice los siguientes pasos adicionales:
   a) Emita el mandato **REFRESH CLUSTER** en el gestor de colas que está añadiendo.

      Este paso es necesario detiene los canales del clúster y entrega a la memoria caché de clúster local un nuevo conjunto de números de secuencia que con toda seguridad están al día dentro del resto del clúster.

      ```
      REFRESH CLUSTER(INVENTORY) REPOS(YES)
      ```

      Nota: Para los clústeres de gran tamaño, el uso del mandato **REFRESH CLUSTER** puede interrumpir el clúster mientras está en curso y, a partir de entonces, de nuevo a intervalos de 27 días cuando los objetos de clúster envían automáticamente actualizaciones de estado a todos los gestores de colas interesados. Consulte La renovación en un clúster grande puede afectar el rendimiento y la disponibilidad del clúster.
   b) Reinicie el canal CLUSSDR.

      Por ejemplo mediante el mandato START CHANNEL.
   c) Reinicie el canal CLUSRCVR.

Resultados

El clúster configurado por esta tarea es el mismo que para “Añadir un gestor de colas a un clúster” en la página 335:

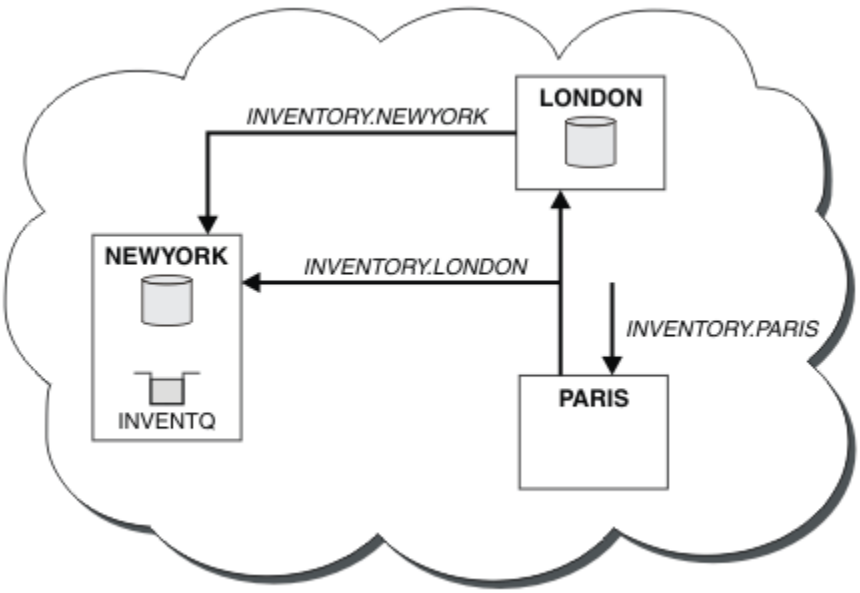


Figura 41. El clúster INVENTORY con tres gestores de colas

Haciendo sólo dos definiciones, una definición CLUSRCVR y una definición CLUSSDR, hemos añadido el gestor de colas PARIS al clúster.

En el gestor de colas PARIS, se inicia el CLUSSDR que contiene la serie +QMNAME+. En el sistema LONDON, IBM MQ resuelve +QMNAME+ en el nombre del gestor de colas (LONDON). A continuación, IBM MQ hace coincidir la definición de un canal denominado INVENTORY.LONDON con la definición correspondiente de CLUSRCVR.

IBM MQ devuelve el nombre de canal resuelto al gestor de colas PARIS. En PARIS, la definición de canal CLUSSDR para el canal llamado INVENTORY.+QMNAME+ se sustituye por una definición CLUSSDR generada internamente para INVENTORY.LONDON. Esta definición contiene el nombre de canal resuelto pero, por lo demás, es la misma que la definición +QMNAME+ que ha realizado. Los repositorios del clúster también se actualizan con la definición de canal con el nombre de canal recién resuelto.

Nota:

1. El canal creado con el nombre +QMNAME+ queda inactivo inmediatamente. No se utiliza nunca para transmitir datos.
2. Las salidas de canal pueden ver el cambio de nombre de canal entre una invocación y la siguiente.

Ahora, el gestor de colas PARIS se informa, en el repositorio en LONDON, que la cola INVENTQ está alojada por el gestor de colas NEWYORK. Cuando una aplicación alojada por el sistema en París intenta transferir mensajes al INVENTQ, PARIS define automáticamente un canal de clúster emisor para conectarse al canal de clúster receptor INVENTORY.NEWYORK. La aplicación puede recibir respuestas cuando su nombre de gestor de colas se especifica como el gestor de colas de destino y se proporciona una cola de respuesta.

Tareas relacionadas

Añadir un gestor de colas a un clúster: colas de transmisión separadas
Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando varias colas de transmisión de clúster.

Referencia relacionada

DEFINE CHANNEL

Añadir un gestor de colas que aloja una cola

Añada otro gestor de colas al clúster, para alojar otra cola INVENTQ. Las solicitudes se envían alternativamente a las colas en cada gestor de colas. No es necesario realizar ningún cambio en el host INVENTQ existente.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- El clúster de INVENTORY se ha configurado como se describe en “Añadir un gestor de colas a un clúster” en la página 335. Contiene tres gestores de colas; LONDON y NEWYORK contienen ambos depósitos completos, PARÍS contiene un depósito parcial. La aplicación de inventario se ejecuta en el sistema de Nueva York, conectada al gestor de colas NEWYORK. La aplicación se activa con la llegada de mensajes a la cola INVENTQ.
- Se está abriendo una nueva tienda en Toronto. Para proporcionar capacidad adicional, desea ejecutar la aplicación de inventario en el sistema de Toronto y en el de Nueva York.
- Existe conectividad de red entre los cuatro sistemas.
- El protocolo de red es TCP.

Nota: El gestor de colas TORONTO contiene sólo un depósito parcial. Si desea añadir un gestor de colas de depósito completo a un clúster, consulte "Trasladar un depósito completo a otro gestor de colas" en la página 347.

Acerca de esta tarea

Siga estos pasos para añadir un gestor de colas que aloja una cola.

Procedimiento

1. Decida a qué repositorio completo hace referencia primero TORONTO.

   Cada gestor de colas de un clúster debe hacer referencia a uno de los dos repositorios completos. El gestor de colas recopila información sobre el clúster de un repositorio completo y así crea su propio repositorio parcial. No tiene mucha importancia qué repositorio elija. En este ejemplo, elegimos NEWYORK. Una vez que el nuevo gestor de colas se ha unido al clúster, se comunica con los dos repositorios.
2. Defina el canal CLUSRCVR.

   Cada gestor de colas de un clúster tiene que definir un canal de clúster receptor en el que pueda recibir mensajes. En TORONTO, defina un canal CLUSRCVR:

   ```
   DEFINE CHANNEL(INVENTORY.TORONTO) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(TORONTO.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-receiver channel for TORONTO')
   ```

   El gestor de colas TORONTO anuncia su disponibilidad para recibir mensajes de otros gestores de colas en el clúster INVENTORY mediante su canal de clúster receptor.
3. Defina un canal CLUSSDR en el gestor de colas TORONTO.

   Cada gestor de colas de un clúster debe definir un canal de clúster emisor en el que pueda enviar mensajes a su primer repositorio completo. En este ejemplo, elija NEWYORK. TORONTO necesita la siguiente definición:

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from TORONTO to repository at NEWYORK')
   ```
4. Opcional: Si está añadiendo a un clúster un gestor de colas que anteriormente se había eliminado del mismo clúster, compruebe que se muestre ahora como un miembro del clúster. Si no es así, realice los siguientes pasos adicionales:
   a) Emita el mandato **REFRESH CLUSTER** en el gestor de colas que está añadiendo.

      Este paso es necesario detiene los canales del clúster y entrega a la memoria caché de clúster local un nuevo conjunto de números de secuencia que con toda seguridad están al día dentro del resto del clúster.

      ```
      REFRESH CLUSTER(INVENTORY) REPOS(YES)
      ```

      Nota: Para los clústeres de gran tamaño, el uso del mandato **REFRESH CLUSTER** puede interrumpir el clúster mientras está en curso y, a partir de entonces, de nuevo a intervalos de 27 días cuando los objetos de clúster envían automáticamente actualizaciones de estado a todos los gestores de colas interesados. Consulte La renovación en un clúster grande puede afectar el rendimiento y la disponibilidad del clúster.
   b) Reinicie el canal CLUSSDR.

      Por ejemplo mediante el mandato START CHANNEL.
   c) Reinicie el canal CLUSRCVR.
5. Revise la aplicación de inventario para ver si tiene afinidades de mensajes.

Antes de continuar, asegúrese de que la aplicación de inventario no tiene ninguna dependencia de la secuencia de proceso de mensajes e instale la aplicación en el sistema de Toronto.

6. Defina la cola de clúster INVENTQ.

   La cola INVENTQ, que ya está alojada por el gestor de colas NEWYORK, también se va a alojar en TORONTO. Defínala en el gestor de colas TORONTO como se indica a continuación:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

Resultados

La Figura 42 en la página 344 muestra el clúster INVENTORY configurado por esta tarea.

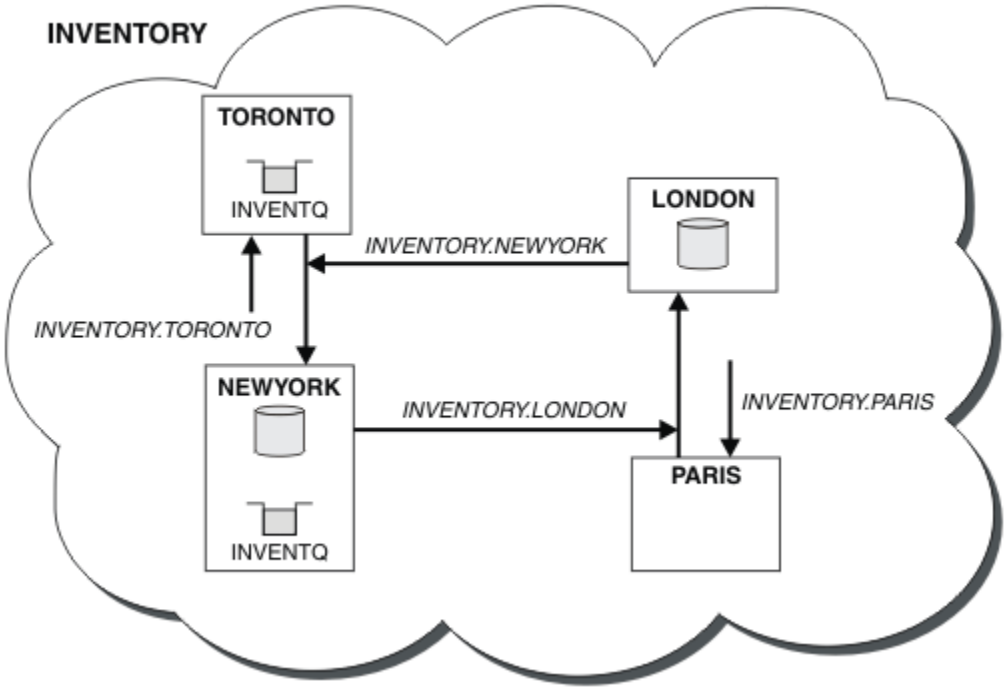


Figura 42. El clúster INVENTORY con cuatro gestores de colas

La cola INVENTQ y la aplicación de inventario ahora están alojadas en dos gestores de colas del clúster. Esto aumenta su disponibilidad, acelera el rendimiento de los mensajes y permite distribuir la carga de trabajo entre los dos gestores de colas. Los mensajes transferidos a INVENTQ por TORONTO o NEWYORK son manejados por la instancia en el gestor de colas local siempre que sea posible. Los mensajes transferidos por LONDON o PARIS se direccionan alternativamente a TORONTO o NEWYORK, para equilibrar la carga de trabajo.

Esta modificación en el clúster se ha llevado a cabo sin que haya tenido que modificar las definiciones en los gestores de colas NEWYORK, LONDON y PARIS. Los repositorios completos en estos gestores de colas se actualizan automáticamente con la información que necesitan para poder enviar mensajes a INVENTQ en TORONTO. La aplicación de inventario sigue funcionando si uno de los gestores de colas NEWYORK o TORONTO queda no disponible, y tiene suficiente capacidad. La aplicación de inventario debe poder funcionar correctamente si está alojada en ambas ubicaciones.

Como puede ver en el resultado de esta tarea, puede tener la misma aplicación ejecutándose en más de un gestor de colas. Puede utilizar la agrupación en clúster para distribuir la carga de trabajo de manera uniforme.

Es posible que una aplicación no pueda procesar registros en ambas ubicaciones. Por ejemplo, suponga que decide añadir una consulta de cuenta de cliente y actualizar la aplicación que se ejecuta en LONDON y NEWYORK. Un registro de cuenta sólo se puede mantener en un lugar. Puede decidir controlar la

distribución de las solicitudes utilizando una técnica de particionamiento de datos. Puede dividir la distribución de los registros. Puede disponer que la mitad de los registros, por ejemplo los números de cuenta 00000 a 49999, se mantengan en LONDON. La otra mitad, en el rango de 50000 a 99999, se mantienen en NEWYORK. A continuación podría escribir un programa de salida de carga de trabajo de clúster para examinar el campo de cuenta en todos los mensajes, y direccionar los mensajes al gestor de colas adecuado.

Qué hacer a continuación

z/OS Ahora que ha completado todas las definiciones, si todavía no lo ha hecho, inicie el iniciador de canal en IBM MQ for z/OS.

En todas las plataformas, inicie un programa de escucha en el gestor de colas TORONTO. El programa de escucha espera peticiones de red entrantes e inicia el canal de clúster receptor cuando es necesario.

z/OS Adding a queue sharing group to existing clusters

Add a queue sharing group on z/OS to existing clusters.

Before you begin

Note:

1. For changes to a cluster to be propagated throughout the cluster, at least one full repository must always be available. Ensure that your repositories are available before starting this task.
2. Queue sharing groups are supported only on IBM MQ for z/OS. This task is not applicable to other platforms.

Scenario:

- The INVENTORY cluster has been set up as described in “Configurar un nuevo clúster” on page 324. It contains two queue managers, LONDON and NEWYORK.
- You want to add a queue sharing group to this cluster. The group, QSGP, comprises three queue managers, P1, P2, and P3. They share an instance of the INVENTQ queue, which is to be defined by P1.

About this task

Follow these steps to add new queue managers that host a shared queue.

Procedure

1. Decide which full repository the queue managers refer to first.

   Every queue manager in a cluster must refer to one or other of the full repositories. It gathers information about the cluster from a full repository and so builds up its own partial repository. It is of no particular significance which full repository you choose. In this example, choose NEWYORK. Once the queue sharing group has joined the cluster, it communicates with both of the full repositories.
2. Define the CLUSRCVR channels.

   Every queue manager in a cluster needs to define a cluster-receiver channel on which it can receive messages. On P1, P2, and P3, define:

   ```
   DEFINE CHANNEL(INVENTORY.Pn) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(Pn.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-receiver channel for sharing queue manager')
   ```

   The cluster-receiver channel advertises the availability of each queue manager to receive messages from other queue managers in the cluster INVENTORY.
3. Define a CLUSSDR channel for the queue sharing group.

Every member of a cluster needs to define one cluster-sender channel on which it can send messages to its first full repository. In this case we have chosen NEWYORK. One of the queue managers in the queue sharing group needs the following group definition. The definition ensures that every queue manager has a cluster-sender channel definition.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) QSGDISP(GROUP)
DESCR('Cluster-sender channel to repository at NEWYORK')
```

4. Define the shared queue.

   Define the queue INVENTQ on P1 as follows:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY) QSGDISP(SHARED) CFSTRUCT(STRUCTURE)
```

   Start the channel initiator and a listener program on the new queue manager. The listener program listens for incoming network requests and starts the cluster-receiver channel when it is needed.

Results

Figure 43 on page 346 shows the cluster set up by this task.

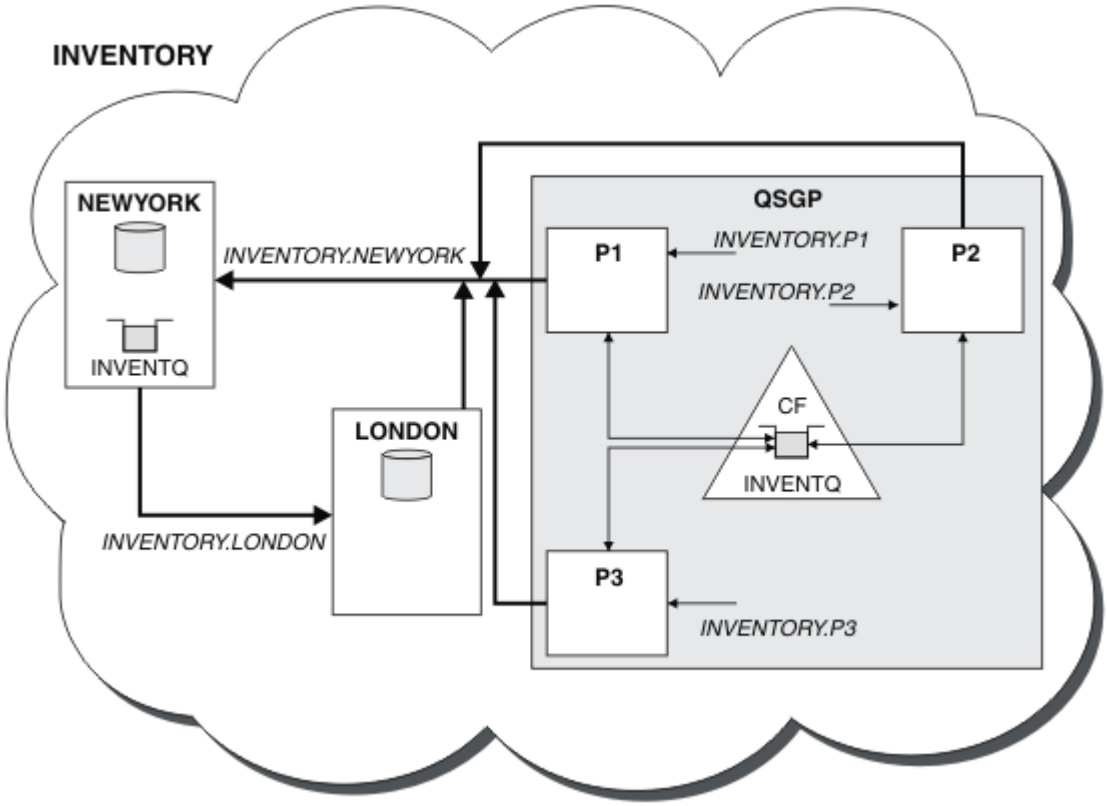


Figure 43. Cluster and queue sharing group

Now messages put on the INVENTQ queue by LONDON are routed alternately around the four queue managers advertised as hosting the queue.

What to do next

A benefit of having members of a queue sharing group host a cluster queue is any member of the group can reply to a request. In this case perhaps P1 becomes unavailable after receiving a message on the shared queue. Another member of the queue sharing group can reply instead.

Trasladar un depósito completo a otro gestor de colas

Traslade un repositorio completo de un gestor de colas a otro, creando el nuevo repositorio a partir de la información contenida en el segundo repositorio.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- El clúster de INVENTORY se ha configurado como se describe en "Añadir un gestor de colas a un clúster" en la página 335.
- Por razones de negocio, ahora desea eliminar el depósito completo del gestor de colas LONDON y sustituirlo por un depósito completo en el gestor de colas PARIS. El gestor de colas NEWYORK va a seguir manteniendo un depósito completo.

Acerca de esta tarea

Siga estos pasos para trasladar un repositorio completo a otro gestor de colas.

Procedimiento

1. Modifique PARIS para que sea un gestor de colas de repositorio completo.

   En PARIS, emita el siguiente mandato:

   ```
   ALTER QMGR REPOS(INVENTORY)
   ```

2. Añada un canal CLUSSDR en PARIS

   PARIS tiene actualmente un canal de clúster emisor que apunta a LONDON. LONDON ya no va a mantener un repositorio completo para el clúster. PARIS debe tener un nuevo canal de clúster emisor que apunte a NEWYORK, donde ahora se mantiene el otro repositorio completo.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from PARIS to repository at NEWYORK')
   ```

3. Defina un canal CLUSSDR en NEWYORK que apunte a PARIS

   Actualmente NEWYORK tiene un canal de clúster emisor que apunta a LONDON. Ahora que el otro repositorio completo se ha trasladado a PARIS, debe añadir un nuevo canal de clúster emisor en NEWYORK que apunte a PARIS.

   ```
   DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from NEWYORK to repository at PARIS')
   ```

   Cuando se añade el canal de clúster emisor a PARIS, PARIS se informa sobre el clúster NEWYORK. Crea su propio repositorio completo utilizando la información de NEWYORK.

4. Compruebe que el gestor de colas PARIS tiene ahora un repositorio completo

   Compruebe que el gestor de colas PARIS ha creado su propio repositorio completo a partir del repositorio completo en el gestor de colas NEWYORK. Emita los siguientes mandatos:

   ```
   DIS QCLUSTER(*) CLUSTER (INVENTORY)
   DIS CLUSQMGR(*) CLUSTER (INVENTORY)
   ```

Compruebe que estos mandatos muestran detalles de los mismos recursos en este clúster que en NEWYORK.

Nota: Si el gestor de colas NEWYORK no está disponible, la creación de esta información no se puede completar. No continúe con el paso siguiente hasta que la tarea se haya completado.

5. Modifique la definición de gestor de colas en LONDON

   Por último, modifique el gestor de colas en LONDON para que ya no contenga un repositorio completo para el clúster. En LONDON, emita el mandato:

   ```
   ALTER QMGR REPOS(' ')
   ```

   El gestor de colas ya no recibe ninguna información del clúster. Después de 30 días, la información que está almacenada en su repositorio completo caduca. El gestor de colas LONDON ahora crea su propio repositorio parcial.

6. Elimine o cambie las definiciones pendientes.

   Cuando esté seguro de que la nueva disposición del clúster funciona según lo esperado, elimine o cambie las definiciones de CLUSSDR definidos manualmente que ya no son correctas.

   - En el gestor de colas PARIS, debe detener y suprimir el canal de clúster emisor a LONDON, y luego emitir el mandato start channel para que el clúster pueda utilizar de nuevo los canales automáticos:

     ```
     STOP CHANNEL(INVENTORY.LONDON)
     DELETE CHANNEL(INVENTORY.LONDON)
     START CHANNEL(INVENTORY.LONDON)
     ```

   - En el gestor de colas NEWYORK, debe detener y suprimir el canal de clúster emisor a LONDON, y luego emitir el mandato start channel para que el clúster pueda utilizar de nuevo los canales automáticos:

     ```
     STOP CHANNEL(INVENTORY.LONDON)
     DELETE CHANNEL(INVENTORY.LONDON)
     START CHANNEL(INVENTORY.LONDON)
     ```

   - Sustituya todos los demás canales de clúster emisor que apunten a LONDON en todos los gestores de colas del clúster por canales que apunten a NEWYORK o PARIS. Después de suprimir un canal, emita siempre el mandato **start channel**, de modo que el clúster pueda volver a utilizar los canales automáticos. En este pequeño ejemplo, no hay ningún otro. Para comprobar si hay algún otro que ha olvidado, emita el mandato DISPLAY CHANNEL desde cada gestor de colas, especificando TYPE(CLUSSDR). Por ejemplo:

     ```
     DISPLAY CHANNEL(*) TYPE(CLUSSDR)
     ```

   Es importante que realice esta tarea lo antes posible después de trasladar el repositorio completo de LONDON a PARIS. En el tiempo transcurrido antes de que realice esta tarea, los gestores de colas que tienen canales CLUSSDR definidos automáticamente llamados INVENTORY.LONDON podrían enviar solicitudes de información utilizando este canal.

   Una vez que LONDON ha dejado de ser un repositorio completo, si recibe estas solicitudes, grabará mensajes de error en su registro de errores de gestor de colas. Los ejemplos siguientes muestran qué mensajes de error se pueden ver en LONDON:

   - AMQ9428: Unexpected publication of a cluster queue object received
   - AMQ9432: Query received by a non-repository queue manager

   El gestor de colas LONDON no responde a las solicitudes de información porque ya no es un repositorio completo. Los gestores de colas que solicitan información de LONDON deben depender de NEWYORK para obtener información del clúster hasta que sus definiciones de CLUSSDR definidos manualmente se corrijan para que apunten a PARIS. Esta situación no debe tolerarse como una configuración válida a largo plazo.

Resultados

Figura 44 en la página 349 muestra el clúster configurado por esa tarea.

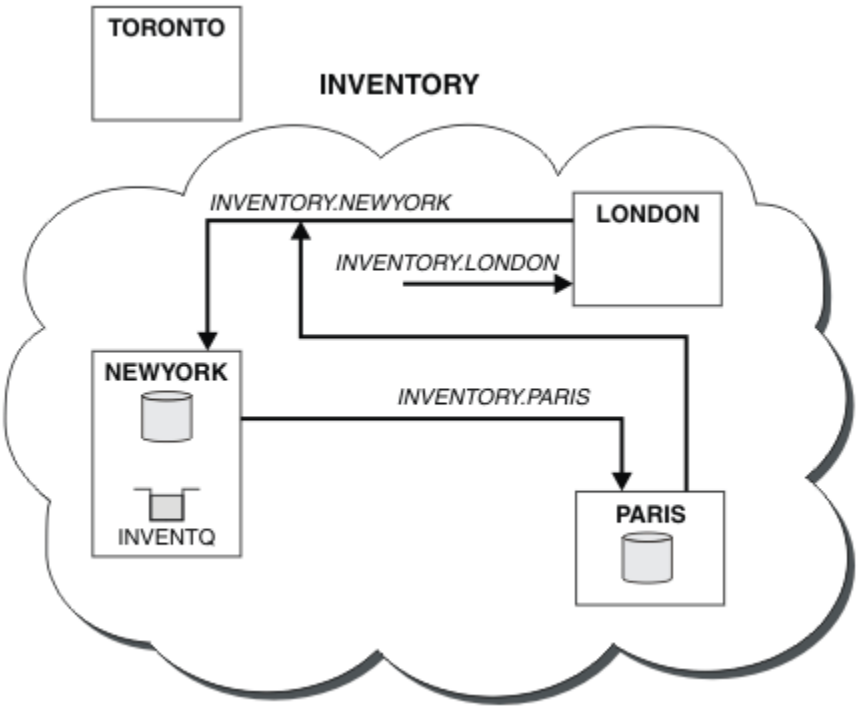


Figura 44. El clúster INVENTORY con el depósito completo trasladado a PARIS

Convertir una red existente en un clúster

Convierta una red existente de colas distribuidas en un clúster y añada un gestor de colas adicional para aumentar la capacidad.

Antes de empezar

En "Configurar un nuevo clúster" en la página 324 a "Trasladar un depósito completo a otro gestor de colas" en la página 347, ha creado y ampliado un nuevo clúster. Las dos tareas siguientes exploran un enfoque distinto: el de la conversión de una red existente de gestores de colas en un clúster.

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- Ya existe una red vigente de IBM MQ, que conecta las sucursales a nivel nacional de una cadena de tiendas. Tiene una estructura en estrella: todos los gestores de colas están conectados a un gestor de colas central. El gestor de colas central está en el sistema en el que se ejecuta la aplicación de inventario. La aplicación se activa con la llegada de mensajes a la cola INVENTQ, para la que cada gestor de colas tiene una definición de cola remota.

  Esta red se ilustra en la Figura 45 en la página 350.

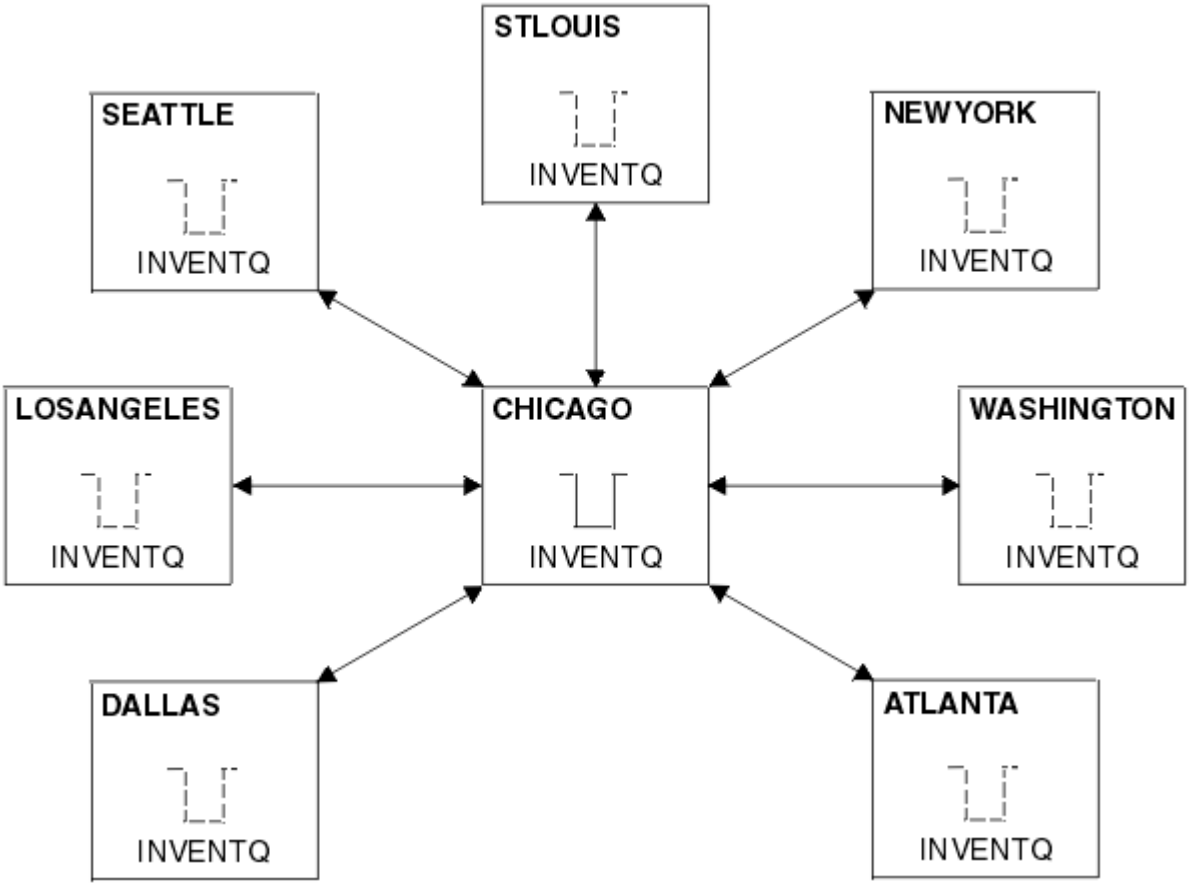


Figura 45. Una red en estrella

- Para facilitar la administración, va a convertir esta red en un clúster y va a crear otro gestor de colas en el sitio central para compartir la carga de trabajo.

  El nombre del clúster es CHNSTORE.

  Nota: El nombre de clúster CHNSTORE se ha seleccionado para permitir que se creen nombres de canal de clúster receptor utilizando nombres con el formato *cluster_name. queue_manager_name* que no superan la longitud máxima de 20 caracteres, por ejemplo CHNSTORE.WASHINGTON.
- Los dos gestores de colas centrales van a contener depósitos completos y ser accesibles a la aplicación de inventario.
- La aplicación de inventario se activará con la llegada de mensajes a la cola INVENTQ alojada por cualquiera de los dos gestores de colas centrales.
- La aplicación de inventario va a ser la única aplicación que se ejecute en paralelo y a la que pueda acceder más de un gestor de colas. Todas las demás aplicaciones seguirán ejecutándose como antes.
- Todas las sucursales tienen conectividad de red con los dos gestores de colas centrales.
- El protocolo de red es TCP.

Acerca de esta tarea

Siga estos pasos para convertir una red existente en un clúster.

Procedimiento

1. Revise la aplicación de inventario para ver si tiene afinidades de mensajes.

   Antes de continuar, asegúrese de que la aplicación puede manejar afinidades de mensajes. Las afinidades de mensajes son la relación entre mensajes conversacionales que se intercambian entre dos aplicaciones, en la que los mensajes deben procesarse mediante un gestor de colas determinado

o en una secuencia determinada. Para obtener más información sobre las afinidades de mensajes, consulte: “Manejo de las afinidades de mensajes” en la página 429

2. Modifique los dos gestores de colas centrales para hacer que sean gestores de colas de repositorio completo.

   Los dos gestores de colas CHICAGO y CHICAGO2 están en el centro de esta red. Ha decidido concentrar toda la actividad asociada con el clúster de cadena de tiendas en esos dos gestores de colas. Además de la aplicación de inventario y las definiciones para la cola INVENTQ, desea que estos gestores de colas alojen los dos repositorios completos para el clúster. En cada uno de los dos gestores de colas, emita el siguiente mandato:

   ```
   ALTER QMGR REPOS(CHNSTORE)
   ```

3. Defina un canal CLUSRCVR en cada gestor de colas.

   En cada gestor de colas del clúster, defina un canal de clúster receptor y un canal de clúster emisor. No importa qué canal defina primero.

   Cree una definición CLUSRCVR para anunciar cada gestor de colas, su dirección de red y otra información, al clúster. Por ejemplo, en el gestor de colas ATLANTA:

   ```
   DEFINE CHANNEL(CHNSTORE.ATLANTA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(ATLANTA.CHSTORE.COM) CLUSTER(CHNSTORE)
   DESCR('Cluster-receiver channel')
   ```

4. Defina un canal CLUSSDR en cada gestor de colas

   Cree una definición CLUSSDR en cada gestor de colas para enlazar ese gestor de colas a cualquiera de los dos gestores de colas de repositorio completo. Por ejemplo, podría enlazar ATLANTA a CHICAGO2:

   ```
   DEFINE CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(CHICAGO2.CHSTORE.COM) CLUSTER(CHNSTORE)
   DESCR('Cluster-sender channel to repository queue manager')
   ```

5. Instale la aplicación de inventario en CHICAGO2.

   Ya tiene la aplicación de inventario en el gestor de colas CHICAGO. Ahora tiene que hacer una copia de esta aplicación en el gestor de colas CHICAGO2.

6. Defina la cola INVENTQ en los gestores de colas centrales.

   En CHICAGO, modifique la definición de cola local correspondiente a la cola INVENTQ para hacer que la cola esté disponible para el clúster. Emita el mandato:

   ```
   ALTER QLOCAL(INVENTQ) CLUSTER(CHNSTORE)
   ```

   En CHICAGO2, cree una definición para la misma cola:

   ```
   DEFINE QLOCAL(INVENTQ) CLUSTER(CHNSTORE)
   ```

   z/OS En z/OS, puede utilizar la opción MAKEDEF de la función COMMAND de **CSQUTIL** para hacer una copia exacta en CHICAGO2 de la cola INVENTQ en CHICAGO.

   Al hacer estas definiciones, se envía un mensaje a los repositorios completos en CHICAGO y CHICAGO2 y se actualiza la información que contienen. Cuando el gestor de colas coloca un mensaje en la cola INVENTQ, descubre en los repositorios completos que hay la posibilidad de elegir entre varios destinos para los mensajes.

7. Compruebe que los cambios realizados en el clúster se han propagado.

   Compruebe que las definiciones que ha creado en el paso anterior se han propagado por el clúster. Emita el siguiente mandato en un gestor de colas de repositorio completo:

```
DIS QCLUSTER(INVENTQ)
```

Añadir un clúster nuevo interconectado

Añada un nuevo clúster que comparta algunos gestores de colas con un clúster existente.

Antes de empezar

Nota:

1. Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.
2. Antes de iniciar esta tarea, debe comprobar si hay conflictos de nombres de cola y entender las consecuencias. Es posible que tenga que renombrar una cola, o configurar alias de cola antes de poder continuar.

Escenario:

- Se ha configurado un clúster de IBM MQ tal como se describe en "Convertir una red existente en un clúster" en la página 349.
- Se va a implementar un nuevo clúster llamado MAILORDER. Este clúster consta de cuatro de los gestores de colas que están en el clúster CHNSTORE: CHICAGO, CHICAGO2, SEATTLE y ATLANTA, y dos gestores de colas adicionales: HARTFORD y OMAHA. La aplicación MAILORDER se ejecuta en el sistema en Omaha, conectada al gestor de colas OMAHA. Se activa cuando los otros gestores de colas del clúster colocan mensajes en la cola MORDERQ.
- Los depósitos completos para el clúster MAILORDER se mantienen en los dos gestores de colas CHICAGO y CHICAGO2.
- El protocolo de red es TCP.

Acerca de esta tarea

Siga estos pasos para añadir un clúster nuevo interconectado.

Procedimiento

1. Cree una lista de nombres de los nombres de clúster.

   Los gestores de colas de repositorio completo en CHICAGO y CHICAGO2 ahora van a mantener los repositorios completos para los dos clústeres, CHNSTORE y MAILORDER. En primer lugar, cree una lista de nombres que contenga los nombres de los clústeres. Defina la lista de nombres en CHICAGO y CHICAGO2, de la manera siguiente:

   ```
   DEFINE NAMELIST(CHAINMAIL)
   DESCR('List of cluster names')
   NAMES(CHNSTORE, MAILORDER)
   ```

2. Modifique las dos definiciones de gestor de colas.

   Ahora, modifique las dos definiciones de gestor de colas en CHICAGO y CHICAGO2. Actualmente estas definiciones muestran que los gestores de colas contienen repositorios completos para el clúster CHNSTORE. Cambie esa definición para que muestre que los gestores de colas contienen repositorios completos para todos los clústeres que aparece en la lista de nombres CHAINMAIL. Modifique las definiciones de los gestores de colas CHICAGO y CHICAGO2:

   ```
   ALTER QMGR REPOS(' ') REPOSNL(CHAINMAIL)
   ```

3. Modifique los canales CLUSRCVR en CHICAGO y CHICAGO2.

Las definiciones de canal CLUSRCVR en CHICAGO y CHICAGO2 muestran que los canales están disponibles en el clúster CHNSTORE. Debe cambiar la definición de clúster receptor para que muestre que los canales están disponibles para todos los clústeres que aparecen en la lista de nombres CHAINMAIL. Cambie la definición del clúster receptor en CHICAGO:

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSRCVR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

En CHICAGO2, emita el mandato:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSRCVR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

4. Modifique los canales CLUSSDR en CHICAGO y CHICAGO2.

   Cambie las dos definiciones de canal CLUSSDR para añadir la lista de nombres. En CHICAGO, emita el mandato:

   ```
   ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

   En CHICAGO2, emita el mandato:

   ```
   ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSSDR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

5. Cree una lista de nombres en SEATTLE y ATLANTA.

   Puesto que SEATTLE y ATLANTA van a ser miembros de más de un clúster, debe crear una lista de nombres que contenga los nombres de los clústeres. Defina la lista de nombres en SEATTLE y ATLANTA de la siguiente manera:

   ```
   DEFINE NAMELIST(CHAINMAIL)
   DESCR('List of cluster names')
   NAMES(CHNSTORE, MAILORDER)
   ```

6. Modifique los canales CLUSRCVR en SEATTLE y ATLANTA.

   Las definiciones de canal CLUSRCVR en SEATTLE y ATLANTA muestran que los canales están disponibles en el clúster CHNSTORE. Cambie las definiciones de clúster receptor para que muestren que los canales están disponibles para todos los clústeres que aparecen en la lista de nombres CHAINMAIL. En SEATTLE, emita el mandato:

   ```
   ALTER CHANNEL(CHNSTORE.SEATTLE) CHLTYPE(CLUSRCVR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

   En ATLANTA, emita el mandato:

   ```
   ALTER CHANNEL(CHNSTORE.ATLANTA) CHLTYPE(CLUSRCVR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

7. Modifique los canales CLUSSDR en SEATTLE y ATLANTA.

   Cambie las dos definiciones de canal CLUSSDR para añadir la lista de nombres. En SEATTLE, emita el mandato:

   ```
   ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSSDR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

En ATLANTA, emita el mandato:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

8. Defina canales CLUSRCVR y CLUSSDR en HARTFORD y OMAHA.

   En los dos nuevos gestores de colas, HARTFORD y OMAHA, defina canales de clúster receptor y de clúster emisor. No importa en qué orden cree las definiciones. En HARTFORD, escriba:

```
DEFINE CHANNEL(MAILORDER.HARTFORD) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(HARTFORD.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-receiver channel for HARTFORD')

DEFINE CHANNEL(MAILORDER.CHICAGO) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-sender channel from HARTFORD to repository at CHICAGO')
```

   En OMAHA, escriba:

```
DEFINE CHANNEL(MAILORDER.OMAHA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(OMAHA.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-receiver channel for OMAHA')

DEFINE CHANNEL(MAILORDER.CHICAGO) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-sender channel from OMAHA to repository at CHICAGO')
```

9. Defina la cola MORDERQ en OMAHA.

   El paso final para completar esta tarea es definir la cola MORDERQ en el gestor de colas OMAHA. En OMAHA, escriba:

```
DEFINE QLOCAL(MORDERQ) CLUSTER(MAILORDER)
```

10. Compruebe que los cambios realizados en el clúster se han propagado.

    Compruebe que las definiciones que ha creado en los pasos anteriores se han propagado por el clúster. Emita los siguientes mandatos en un gestor de colas de repositorio completo:

```
DIS QCLUSTER (MORDERQ)
DIS CLUSQMGR
```

11.

Resultados

El clúster configurado por esta tarea se muestra en la Figura 46 en la página 355.

Ahora tenemos dos clústeres que se solapan. Los depósitos completos para los dos clústeres se mantienen en CHICAGO y CHICAGO2. La aplicación de venta por correo que se ejecuta en OMAHA es independiente de la aplicación de inventario que se ejecuta en CHICAGO. Sin embargo, algunos de los gestores de colas que se encuentran en el clúster CHNSTORE también se encuentran en el clúster MAILORDER, por lo que pueden enviar mensajes a cualquiera de las dos aplicaciones. Antes de llevar a cabo esta tarea para solapar los dos clústeres, tenga en cuenta la posibilidad de que existan conflictos de nombres de cola.

Supongamos que en NEWYORK en el clúster CHNSTORE y en OMAHA en el clúster MAILORDER, hay una cola llamada ACCOUNTQ. Si solapa los clústeres y después una aplicación en SEATTLE transfiere un mensaje a la cola ACCOUNTQ, el mensaje puede ir a cualquiera de las dos instancias de la cola ACCOUNTQ.

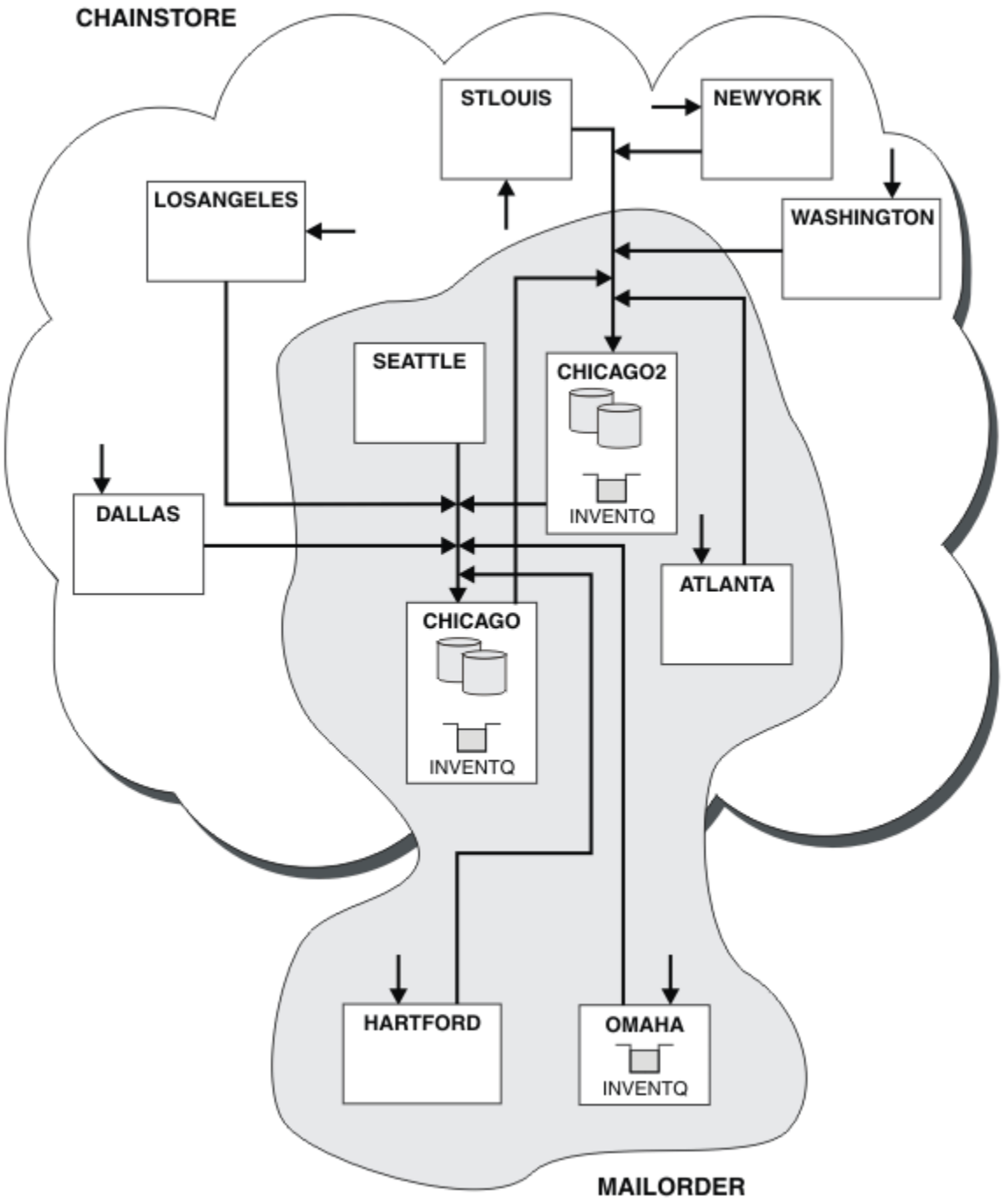


Figura 46. Clústeres interconectados

Qué hacer a continuación

Supongamos que decide fusionar el clúster `MAILORDER` con el clúster `CHNSTORE` para formar un clúster grande llamado `CHNSTORE`.

Para fusionar el clúster `MAILORDER` con el clúster `CHNSTORE`, de manera que `CHICAGO` y `CHICAGO2` contengan los repositorios completos:

- Modifique las definiciones del gestor de colas para CHICAGO y CHICAGO2, eliminando el atributo REPOSNL, que especifica la lista de nombres (CHAINMAIL), y sustituyéndolo por un atributo REPOS que especifica el nombre de clúster (CHNSTORE). Por ejemplo:

  ```
  ALTER QMGR(CHICAGO) REPOSNL(' ') REPOS(CHNSTORE)
  ```

- En cada gestor de colas del clúster MAILORDER, modifique todas las definiciones de canal y las definiciones de cola para cambiar el valor del atributo CLUSTER de MAILORDER a CHNSTORE. Por ejemplo, en HARTFORD, escriba:

  ```
  ALTER CHANNEL(MAILORDER.HARTFORD) CLUSTER(CHNSTORE)
  ```

  En OMAHA, escriba:

  ```
  ALTER QLOCAL(MORDERQ) CLUSTER(CHNSTORE)
  ```

- Modifique todas las definiciones que especifican la lista de nombres de clúster CHAINMAIL, es decir, las definiciones de canal CLUSRCVR y CLUSSDR en CHICAGO, CHICAGO2, SEATTLE y ATLANTA, para que especifiquen en su lugar el clúster CHNSTORE.

En este ejemplo, puede ver las ventajas de utilizar listas de nombres. En lugar de modificar las definiciones de gestor de colas para CHICAGO y CHICAGO2, puede modificar el valor de la lista de nombres CHAINMAIL. Del mismo modo, en lugar de modificar las definiciones de canal CLUSRCVR y CLUSSDR en CHICAGO, CHICAGO2, SEATTLE y ATLANTA, puede conseguir el resultado deseado modificando la lista de nombres.

Tareas relacionadas

Eliminar una red de clústeres

Elimine un clúster de una red y restaure la configuración de gestión de colas distribuidas.

Eliminar una red de clústeres

Elimine un clúster de una red y restaure la configuración de gestión de colas distribuidas.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- Un clúster de IBM MQ se ha configurado tal como se describe en “Convertir una red existente en un clúster” en la página 349.
- Este clúster ahora se va a eliminar del sistema. La red de gestores de colas va a seguir funcionando como lo hacía antes de que se implementara el clúster.

Acerca de esta tarea

Siga estos pasos para eliminar una red de clústeres.

Procedimiento

1. Elimine las colas de clúster del clúster CHNSTORE.

   En CHICAGO y CHICAGO2, modifique la definición de cola local para la cola INVENTQ para eliminar la cola del clúster. Emita el mandato:

   ```
   ALTER QLOCAL(INVENTQ) CLUSTER(' ')
   ```

Cuando modifica la cola, la información de los repositorios completos se actualiza y se propaga por todo el clúster. Las aplicaciones activas que utilizan MQOO_BIND_NOT_FIXED, y las aplicaciones que utilizan MQOO_BIND_AS_Q_DEF en las que la cola se ha definido con DEFBIND(NOTFIXED), fallan la próxima vez que se intenta emitir una llamada MQPUT o MQPUT1. Se devuelve el código de razón MQRC_UNKNOWN_OBJECT_NAME.

No es obligatorio realizar primero el Paso 1, pero si no lo hace, entonces llévelo a cabo después del Paso 4.

2. Detenga todas las aplicaciones que tienen acceso a la cola de clúster.

   Detenga todas las aplicaciones que tienen acceso a las colas de clúster. De lo contrario, es posible que la información del clúster permanezca en el gestor de colas local cuando renueve el clúster en el paso 5. Esta información se elimina cuando todas las aplicaciones se han detenido y los canales de clúster se han desconectado.

3. Elimine el atributo de repositorio de los gestores de colas de repositorio completo.

   En CHICAGO y CHICAGO2, modifique las definiciones de gestor de colas para eliminar el atributo de repositorio. Para ello, emita el mandato:

   ```
   ALTER QMGR REPOS(' ')
   ```

   Los gestores de colas informan a los otros gestores de colas del clúster de que ya no contienen los repositorios completos. Cuando los otros gestores de colas reciben esta información, verá un mensaje que indica que el repositorio completo ha finalizado. También verá uno o más mensajes que indican que ya no hay ningún repositorio disponible para el clúster CHNSTORE.

4. Elimine los canales de clúster.

   En CHICAGO, elimine los canales de clúster:

   ```
   ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR) CLUSTER(' ')
   ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSRCVR) CLUSTER(' ')
   ```

   Nota: Es importante emitir primero el mandato CLUSSDR y luego el mandato CLUSRCVR. No emita primero el mandato CLUSRCVR y luego el mandato CLUSSDR. Si hace eso, crea canales pendientes que tienen un estado STOPPED. A continuación, debe emitir un mandato START CHANNEL para recuperar los canales detenidos; por ejemplo, START CHANNEL(CHNSTORE.CHICAGO).

   Verá mensajes que indican que no hay ningún repositorio para el clúster CHNSTORE.

   Si no ha eliminado las colas de clúster como se describe en el Paso 1, hágalo ahora.

5. Detenga los canales de clúster.

   En CHICAGO, detenga los canales de clúster con los siguientes mandatos:

   ```
   STOP CHANNEL(CHNSTORE.CHICAGO2)
   STOP CHANNEL(CHNSTORE.CHICAGO)
   ```

6. Repita los pasos 4 y 5 para cada gestor de colas en el clúster.
7. Detenga los canales de clúster; a continuación elimine todas las definiciones para los canales de clúster y las colas de clúster de cada gestor de colas.
8. Opcional: Borre la información de clúster almacenada en memoria caché conservada por el gestor de colas.

   Si bien los gestores de colas ya no son miembros del clúster, cada uno mantiene una copia almacenada en memoria caché de la información del clúster. Si desea eliminar estos datos, consulte la tarea “Restauración de un gestor de colas al estado previo al clúster” en la página 385.

9. Sustituya las definiciones de cola remota para la cola INVENTQ

   Para que la red pueda seguir funcionando, sustituya la definición de cola remota para la cola INVENTQ en cada gestor de colas.

10. Reorganice el clúster.

    Suprima todas las definiciones de cola o canal que ya no sean necesarias.

Tareas relacionadas

Añadir un clúster nuevo interconectado
Añada un nuevo clúster que comparta algunos gestores de colas con un clúster existente.

Creación de dos clústeres solapados con un gestor de cola de pasarela

Siga las instrucciones de la tarea para crear clústeres solapados con un gestor de colas de pasarela. Utilice los clústeres como punto de inicio para los siguientes ejemplos de aislamiento de mensajes dirigidos a una aplicación de los mensajes dirigidos a otras aplicaciones de un clúster.

Acerca de esta tarea

El ejemplo de configuración de clúster que se utiliza para ilustrar el aislamiento del tráfico de mensajes del clúster se muestra en la Figura 47 en la página 358. El ejemplo se describe en Agrupación en clúster: Aislamiento de aplicaciones utilizando varias colas de transmisión de clúster.

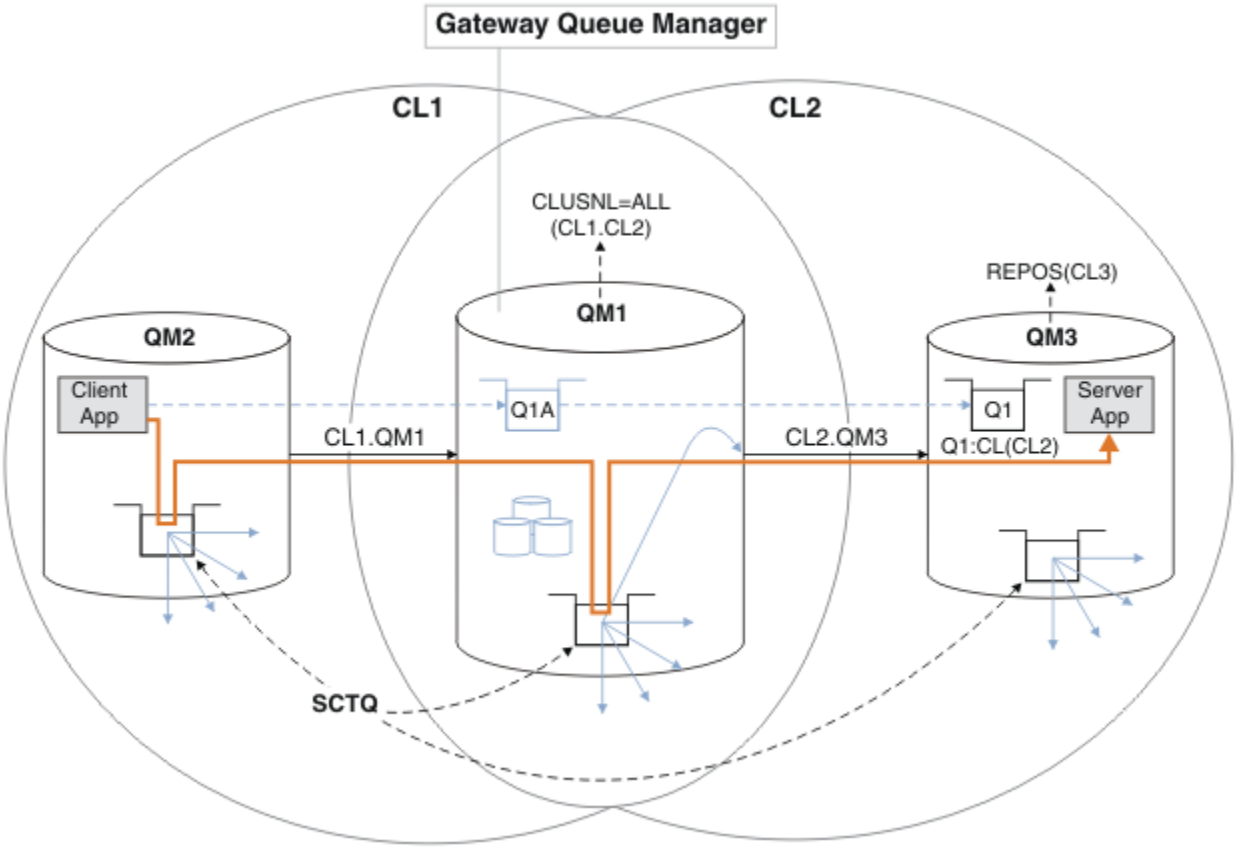


Figura 47. Aplicación cliente-servidor desplegada en una arquitectura en estrella utilizando clústeres de IBM MQ

Para reducir al máximo el número de pasos necesarios para construir el ejemplo, la configuración se mantiene simple, en lugar de realista. El ejemplo puede representar la integración de dos clústeres creados por dos organizaciones independientes. Para obtener un escenario más realista, consulte Agrupación en clúster: Planificación de cómo configurar las colas de transmisión de clúster.

Siga los pasos para crear los clústeres. Los clústeres se utilizan en los siguientes ejemplos de aislamiento del tráfico de mensajes de la aplicación cliente a la aplicación de servidor.

Las instrucciones añaden un par de gestores de colas adicionales para que cada clúster tenga dos repositorios. El gestor de colas de pasarela no se utiliza como repositorio por motivos de rendimiento.

Procedimiento

1. Cree e inicie los gestores de colas QM1, QM2, QM3, QM4, QM5.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE QM n
strmqm QmgrName
```

   Nota: QM4 y QM5 son los repositorios completos de copia de seguridad para los clústeres.

2. Defina e inicie escuchas para cada uno de los gestores de colas.

```
*... On QM n
DEFINE LISTENER(TCP141 n) TRPTYPE(TCP) IPADDR(hostname) PORT(141 n) CONTROL(QMGR) REPLACE
START LISTENER(TCP141 n)
```

3. Cree una lista de nombres de clúster para todos los clústeres.

```
*... On QM1
DEFINE NAMELIST(ALL) NAMES(CL1, CL2) REPLACE
```

4. Haga que QM2 y QM4 sean repositorios completos para CL1, QM3 y QM5 repositorios completos para CL2.

   a) Para CL1:

```
*... On QM2 and QM4
ALTER QMGR REPOS(CL1) DEFCLXQ(SCTQ)
```

   b) Para CL2:

```
*... On QM3 and QM5
ALTER QMGR REPOS(CL2) DEFCLXQ(SCTQ)
```

5. Añada los canales de clúster emisor y clúster receptor para cada gestor de colas y clúster.

   Ejecute los mandatos siguientes en QM2, QM3, QM4 y QM5, donde *c*, *n*y *m* tomen los valores que se muestran en Tabla 27 en la página 359 para cada gestor de colas:

Tabla 27. Valores de parámetros para crear los clústeres 1 y 2

Gestor de colas	**Clúster** ***c***	**Otro depósito** ***n***	**Este depósito** ***m***
QM2	1	4	2
QM4	1	2	4
QM3	2	5	3
QM5	2	3	5

```
*... On QM m
DEFINE CHANNEL(CL c.QM n) CHLTYPE(CLUSSDR) CONNAME('localhost(141 n)') CLUSTER(CL c) REPLACE
DEFINE CHANNEL(CL c.QM m) CHLTYPE(CLUSRCVR) CONNAME('localhost(141 m)') CLUSTER(CL c) REPLACE
```

6. Añada el gestor de colas de pasarela, QM1, a cada uno de los clústeres.

```
*... On QM1
DEFINE CHANNEL(CL1.QM2) CHLTYPE(CLUSSDR) CONNAME('localhost(1412)') CLUSTER(CL1) REPLACE
DEFINE CHANNEL(CL1.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL1) REPLACE
DEFINE CHANNEL(CL2.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL2) REPLACE
DEFINE CHANNEL(CL2.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL2) REPLACE
```

7. Añada la cola local Q1 al gestor de colas QM3 en el clúster CL2.

```
*... On QM3
DEFINE QLOCAL(Q1) CLUSTER(CL2) REPLACE
```

8. Añada el alias de gestor de colas en clúster Q1A al gestor de colas de pasarela.

```
*... On QM1
DEFINE QALIAS(Q1A) CLUSNL(ALL) TARGET(Q1) TARGTYPE(QUEUE) DEFBIND(NOTFIXED) REPLACE
```

Nota: Las aplicaciones que utilizan el alias de gestor de colas en cualquier otro gestor de colas que no sea QM1, deben especificar DEFBIND(NOTFIXED) cuando abren la cola de alias. **DEFBIND** especifica si la información de direccionamiento de la cabecera de mensaje se fija cuando la aplicación abre la cola. Si se establece en el valor predeterminado, OPEN, los mensajes se direccionan a Q1@QM1. Q1@QM1 no existe, por lo que los mensajes de otros gestores de colas terminan en una cola de mensajes no entregados. Al establecer el atributo de cola en DEFBIND(NOTFIXED), aplicaciones como **amqsput**, que toman como valor predeterminado el valor de cola **DEFBIND**, se comportan del modo correcto.

9. Añada las definiciones de alias de gestor de colas de clúster para todos los gestores de colas en clúster al gestor de colas de pasarela, QM1.

```
*... On QM1
DEFINE QREMOTE(QM2) RNAME(' ') RQMNAME(QM2) CLUSNL(ALL) REPLACE
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSNL(ALL) REPLACE
```

Consejo: Las definiciones de alias de gestor de colas en el gestor de colas de pasarela transfieren mensajes que hacen referencia a un gestor de colas en otro clúster; consulte el apartado Alias de gestor de colas de clúster.

Qué hacer a continuación

1. Pruebe la definición de alias de cola enviando un mensaje de QM2 a Q1 en QM3 utilizando la definición de alias de cola Q1A.
   a. Ejecute el programa de ejemplo **amqsput** en QM2 para colocar un mensaje.

```
C:\IBM\MQ>amqsput Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A

Sample AMQSPUT0 end
```

   b. Ejecute el programa de ejemplo **amqsget** para obtener el mensaje de Q1 en QM3

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

2. Pruebe las definiciones de alias de gestor de colas enviando un mensaje de solicitud y recibiendo un mensaje de respuesta en una cola de respuesta dinámica temporal.

   El diagrama muestra el camino que toma el mensaje de respuesta para volver a una cola dinámica temporal, que se llama RQ. La aplicación de servidor, conectada a QM3, abre la cola de respuestas utilizando el nombre del gestor de colas QM2. El nombre del gestor de colas QM2 se define como un alias de gestor de colas en clúster en QM1. QM3 direcciona el mensaje de respuesta a QM1. QM1 direcciona el mensaje a QM2.

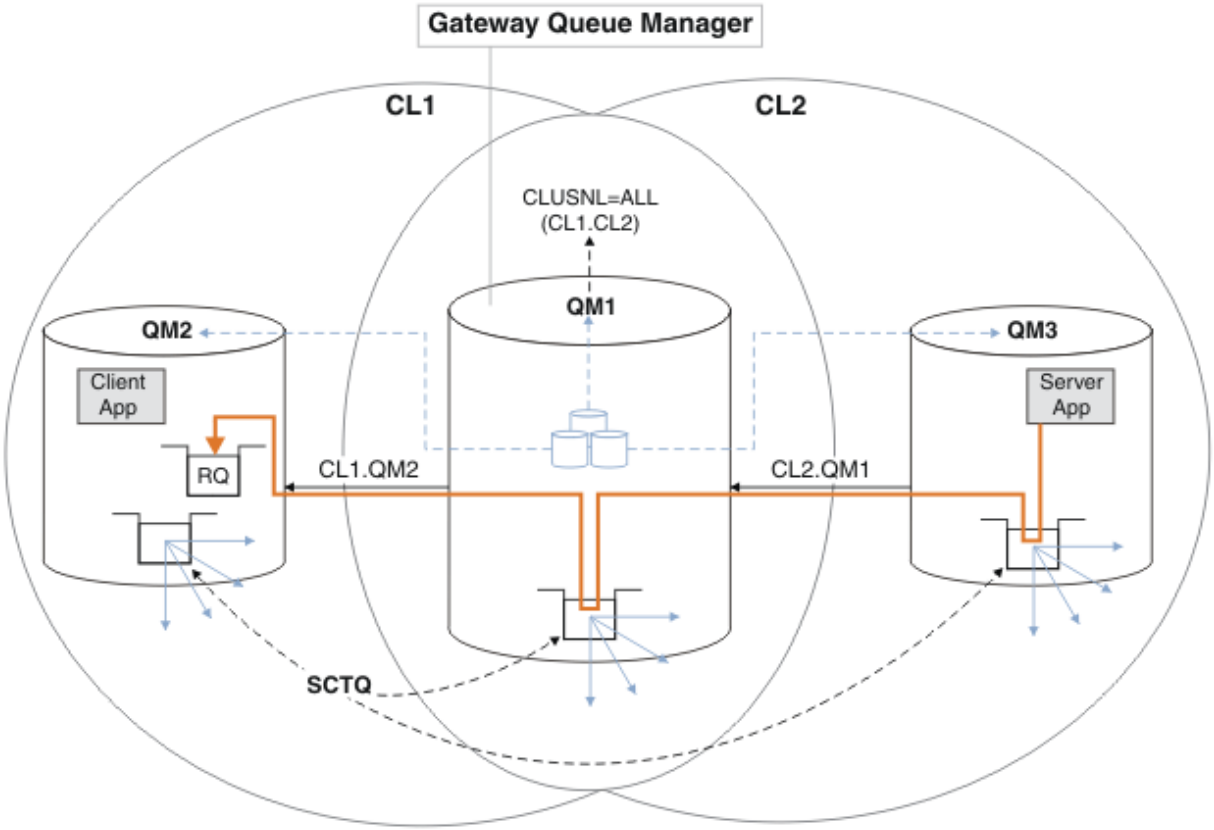


Figura 48. Uso de un alias de gestor de colas para devolver el mensaje de respuesta a un clúster diferente

El modo en que funciona el direccionamiento es el siguiente. Cada gestor de colas de cada clúster tiene una definición de alias de gestor de colas en QM1. Los alias están en clúster en todos los clústeres. Las flechas grises discontinuas de cada uno de los alias a un gestor de colas muestran que cada alias de gestor de colas se resuelve en un gestor de colas real al menos en uno de los clústeres. En este caso, el alias de QM2 se agrupa en el clúster CL1 y CL2, y se resuelve en el gestor de colas real QM2 en CL1. La aplicación de servidor crea el mensaje de respuesta utilizando la respuesta al nombre de cola RQy responde al nombre del gestor de colas QM2. El mensaje se direcciona a QM1 porque la definición de alias del gestor de colas QM2 está definida en QM1 en el clúster CL2 y el gestor de colas QM2 no está en el clúster CL2. Puesto que el mensaje no se puede enviar al gestor de colas de destino, se envía al gestor de colas que tiene la definición de alias.

QM1 coloca el mensaje en la cola de transmisión del clúster en QM1 para transferirlo a QM2. QM1 direcciona el mensaje a QM2 porque la definición de alias del gestor de colas en QM1 para QM2 define QM2 como el gestor de colas de destino real. La definición es no circular, porque las definiciones de alias sólo pueden hacer referencia a definiciones reales; el alias no puede apuntar a sí mismo. QM1 resuelve la definición real, porque tanto QM1 como QM2 están en el mismo clúster, CL1. QM1 averigua la información de conexión de QM2 desde el repositorio para CL1 y direcciona el mensaje a QM2. Para que el mensaje sea redireccionado por QM1, la aplicación de servidor debe haber abierto la cola de respuestas con la opción DEFBIND establecida en MQBND_BIND_NOT_FIXED. Si la aplicación de servidor ha abierto la cola de respuestas con la opción MQBND_BIND_ON_OPEN, el mensaje no se redirecciona y termina en una cola de mensajes no entregados.

a. Cree una cola de solicitud en clúster con un desencadenante en QM3.

```
*... On QM3
DEFINE QLOCAL(QR) CLUSTER(CL2) TRIGGER INITQ(SYSTEM.DEFAULT.INITIATION.QUEUE)
PROCESS(ECHO) REPLACE
```

b. Cree una definición de alias de cola en clúster de QR en el gestor de colas de pasarela, QM1.

```
*... On QM1
DEFINE QALIAS(QRA) CLUSNL(ALL) TARGET(QR) TARGTYPE(QUEUE) DEFBIND(NOTFIXED) REPLACE
```

c. Cree una definición de proceso para iniciar el programa de repetición de ejemplo **amqsech** en QM3.

```
*... On QM3
DEFINE PROCESS(ECHO) APPLICID(AMQSECH) REPLACE
```

d. Cree una cola de modelo en QM2 para el programa de ejemplo **amqsreq** para crear la cola de respuesta temporal dinámica.

```
*... On QM2
DEFINE QMODEL(SYSTEM.SAMPLE.REPLY) REPLACE
```

e. Pruebe la definición de alias de gestor de colas enviando una solicitud de QM2 a QR en QM3 utilizando la definición de alias de cola QRA.

i) Ejecute el programa de supervisor desencadenante en QM3.

```
runmqtrm -m QM3
```

La salida es

```
C:\IBM\MQ>runmqtrm -m QM3
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
01/02/2012  16:17:15: IBM MQ trigger monitor started.

_____________________________________________________
01/02/2012  16:17:15: Waiting for a trigger message
```

ii) Ejecute el programa de ejemplo **amqsreq** en QM2 para colocar una solicitud y esperar una respuesta.

```
C:\IBM\MQ>amqsreq QRA QM2
Sample AMQSREQ0 start
server queue is QRA
replies to 4F2961C802290020
A request message from QM2 to QR on QM3

response <A request message from QM2 to QR on QM3>
no more replies
Sample AMQSREQ0 end
```

Conceptos relacionados

Control de accesos y varias colas de transmisión de clúster

Agrupación en clúster: Aislamiento de aplicaciones utilizando varias colas de transmisión de clúster

Tareas relacionadas

Agrupación en clúster: Planificación de cómo configurar las colas de transmisión de clúster

“Añadir un gestor de colas a un clúster: colas de transmisión separadas” en la página 337

Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando varias colas de transmisión de clúster.

Añadir una definición de cola remota para aislar los mensajes enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la

misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza una definición de cola remota de clúster y un canal emisor y una cola de transmisión distintos.

Antes de empezar

Construya los clústeres solapados que se muestran en Aplicación cliente-servidor desplegada en una arquitectura en estrella utilizando clústeres de IBM MQ en "Creación de dos clústeres solapados con un gestor de cola de pasarela" en la página 358 siguiendo los pasos de dicha tarea.

Acerca de esta tarea

La solución utiliza colas distribuidas para separar los mensajes para la aplicación `Server App` de otro tráfico de mensajes en el gestor de colas de pasarela. Debe definir una definición de cola remota de clúster en `QM1` para desviar los mensajes a una cola de transmisión diferente y a un canal diferente. La definición de cola remota debe incluir una referencia a la cola de transmisión específica que almacena mensajes sólo para `Q1` en `QM3`. En la Figura 49 en la página 363, el alias de la cola de clúster `Q1A` se complementa con una definición de cola remota `Q1R` y una cola de transmisión y un canal emisor añadido.

En esta solución, los mensajes de respuesta se devuelven utilizando la cola común `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

La ventaja de esta solución es que es fácil separar el tráfico para varias colas de destino en el mismo gestor de colas, en el mismo clúster. El inconveniente de la solución es que no se puede utilizar el equilibrio de carga de clúster entre varias copias de `Q1` en distintos gestores de colas. Para superar este inconveniente, consulte "Añadir una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela" en la página 365. También tendrá que gestionar el conmutador de una cola de transmisión a la otra.

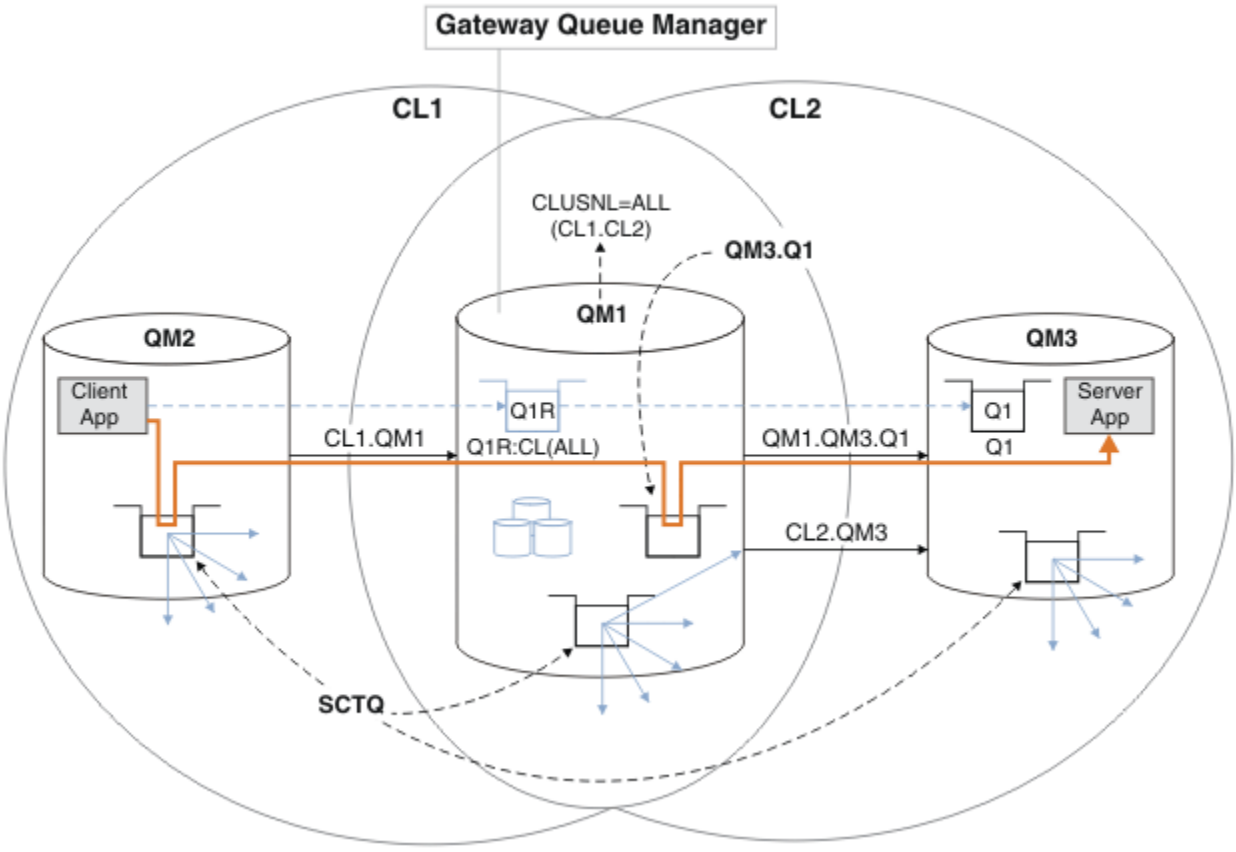


Figura 49. Aplicación cliente-servidor desplegada en una arquitectura de clúster en estrella utilizando definiciones de colas remotas

Procedimiento

1. Cree un canal para separar el tráfico de mensajes para Q1 del gestor de colas de pasarela
   a) Cree un canal emisor en el gestor de colas de pasarela, QM1, para el gestor de colas de destino, QM3.

```
DEFINE CHANNEL(QM1.QM3.Q1) CHLTYPE(SDR) CONNAME(QM3HostName(1413)) XMITQ(QM3.Q1) REPLACE
```

   b) Cree un canal receptor en el gestor de colas de destino, QM3.

```
DEFINE CHANNEL(QM1.QM3.Q1) CHLTYPE(RCVR) REPLACE
```

2. Cree una cola de transmisión en el gestor de colas de pasarela para el tráfico de mensajes a Q1

```
DEFINE QLOCAL(QM3.Q1) USAGE(XMITQ) REPLACE
START CHANNEL(QM1.QM3.Q1)
```

   Al iniciar el canal asociado con la cola de transmisión, se asocia la cola de transmisión con el canal. El canal se inicia automáticamente cuando la cola de transmisión se ha asociado con el canal.
3. Complete la definición de alias de cola de clúster para Q1 en el gestor de colas de pasarela con una definición de cola remota de clúster.

```
DEFINE QREMOTE CLUSNL(ALL) RNAME(Q1) RQMNAME(QM3) XMITQ(QM3.Q1) REPLACE
```

Qué hacer a continuación

Pruebe la configuración enviando un mensaje a Q1 en QM3 de QM2 utilizando la definición de cola remota de clúster Q1R en el gestor de colas de pasarela QM1.

1. Ejecute el programa de ejemplo **amqsput** en QM2 para colocar un mensaje.

```
C:\IBM\MQ>amqsput Q1R QM2
Sample AMQSPUT0 start
target queue is Q1R
Sample request message from QM2 to Q1 using Q1R

Sample AMQSPUT0 end
```

2. Ejecute el programa de ejemplo **amqsget** para obtener el mensaje de Q1 en QM3

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1R>
no more messages
Sample AMQSGET0 end
```

Conceptos relacionados

Agrupación en clúster: Aislamiento de aplicaciones utilizando varias colas de transmisión de clúster

Control de accesos y varias colas de transmisión de clúster

Tareas relacionadas

Añadir una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza una cola de

transmisión de clúster adicional para separar el tráfico de mensajes a un único gestor de colas de un clúster.

Añadir un clúster y una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela
Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza un clúster adicional para aislar los mensajes a una cola de clúster determinada.

Modificar el valor predeterminado para separar colas de transmisión de clúster para aislar el tráfico de mensajes
Puede cambiar el modo predeterminado en que un gestor de colas almacena mensajes para una cola o un tema de clúster en una cola de transmisión. La modificación del valor predeterminado le permite aislar los mensajes de clúster en un gestor de colas de pasarela.

Agrupación en clúster: Planificación de cómo configurar las colas de transmisión de clúster

"Añadir un gestor de colas a un clúster: colas de transmisión separadas" en la página 337
Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando varias colas de transmisión de clúster.

Añadir una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza una cola de transmisión de clúster adicional para separar el tráfico de mensajes a un único gestor de colas de un clúster.

Antes de empezar

1. El gestor de colas de pasarela debe estar en IBM MQ.
2. Construya los clústeres solapados que se muestran en Aplicación cliente-servidor desplegada en una arquitectura en estrella utilizando clústeres de IBM MQ en "Creación de dos clústeres solapados con un gestor de cola de pasarela" en la página 358 siguiendo los pasos de dicha tarea.

Acerca de esta tarea

En el gestor de colas de pasarela, `QM1`, añada una cola de transmisión y establezca su atributo de cola `CLCHNAME`. Establezca como `CLCHNAME` el nombre del canal de clúster receptor en `QM3`; consulte la Figura 50 en la página 366.

Esta solución tiene una serie de ventajas sobre la solución descrita en "Añadir una definición de cola remota para aislar los mensajes enviados desde un gestor de colas de pasarela" en la página 362:

- Requiere menos definiciones adicionales.
- Soporta el equilibrio de carga entre varias copias de la cola de destino, `Q1`, en distintos gestores de colas en el mismo clúster, `CL2`.
- El gestor de colas de pasarela pasa automáticamente a la nueva configuración cuando el canal se reinicia sin perder los mensajes.
- El gestor de colas de pasarela sigue reenviando mensajes en el mismo orden en que los recibió. Lo hace aunque la conmutación tenga lugar con mensajes para la cola `Q1` en `QM3` aún en `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

La configuración para aislar el tráfico de mensajes de clúster en Figura 50 en la página 366 no da como resultado un aislamiento de tráfico tan elevado como la configuración que utiliza colas remotas en "Añadir una definición de cola remota para aislar los mensajes enviados desde un gestor de colas de pasarela" en la página 362. Si el gestor de colas `QM3` en `CL2` aloja varias colas de clúster y aplicaciones de servidor diferentes, todas esas colas comparten el canal de clúster, `CL2.QM3`, conectando `QM1` a `QM3`.

Los flujos adicionales se ilustran en la Figura 50 en la página 366 mediante la flecha gris que representa el tráfico de mensajes de clúster potencial de SYSTEM.CLUSTER.TRANSMIT.QUEUE al canal de clúster emisor CL2.QM3.

El remedio consiste en limitar el gestor de colas para que aloje una sola cola de clúster en un clúster determinado. Si el gestor de colas ya aloja varias colas de clúster, para cumplir esta restricción, debe crear otro gestor de colas, o crear otro clúster; consulte "Añadir un clúster y una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela" en la página 368.

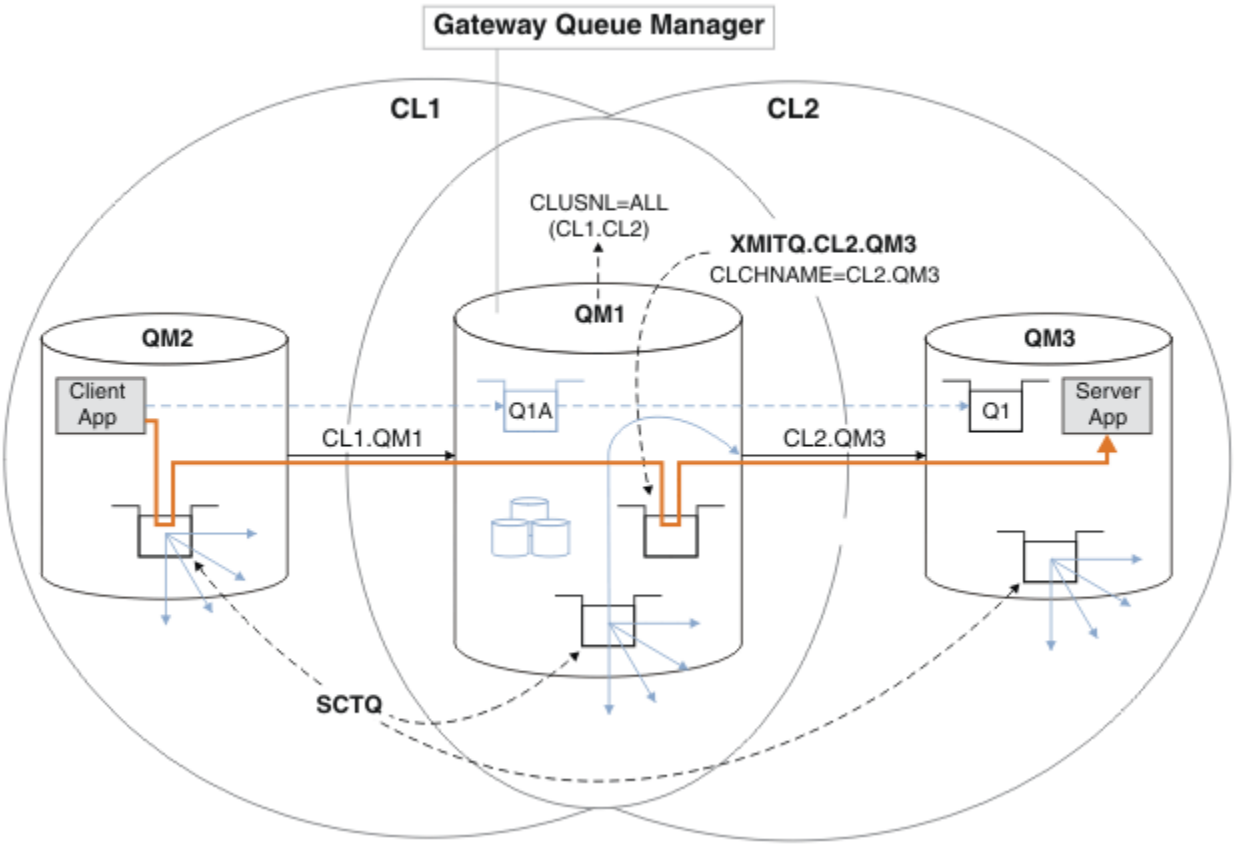


Figura 50. Aplicación cliente-servidor desplegada en una arquitectura en estrella utilizando una cola de transmisión de clúster adicional

Procedimiento

1. Cree una cola de transmisión de clúster adicional para el canal de clúster emisor CL2.QM3 en el gestor de colas de pasarela, QM1.

```
 *... on QM1
DEFINE QLOCAL(XMITQ.CL2.QM3) USAGE(XMITQ) CLCHNAME(CL2.QM3)
```

2. Pase a utilizar la cola de transmisión, XMITQ.CL2.QM3.
   a) Detenga el canal de clúster emisor CL2.QM3.

```
*... On QM1
STOP CHANNEL(CL2.QM3)
```

   La respuesta es que el mandato se ha aceptado:

```
AMQ8019: Stop IBM MQ channel accepted.
```

b) Compruebe que el canal CL2.QM3 se haya detenido

Si el canal no se detiene, puede volver a ejecutar el mandato **STOP CHANNEL** con la opción FORCE . Un ejemplo de definición de la opción FORCE sería si el canal no se detiene y no se puede reiniciar el otro gestor de colas para sincronizar el canal.

```
*... On QM1
start
```

La respuesta es un resumen del estado de canal

```
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)              CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))      CURRENT
RQMNAME(QM3)                  STATUS(STOPPED)
SUBSTATE(MQGET)               XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

c) Inicie el canal, CL2.QM3.

```
*... On QM1
START CHANNEL(CL2.QM3)
```

La respuesta es que el mandato se ha aceptado:

```
AMQ8018: Start IBM MQ channel accepted.
```

d) Compruebe el canal iniciado.

```
*... On QM1
DISPLAY CHSTATUS(CL2.QM3)
```

La respuesta es un resumen del estado de canal:

```
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)                    CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))            CURRENT
RQMNAME(QM3)                        STATUS(RUNNING)
SUBSTATE(MQGET)                     XMITQ(XMITQ.CL2.QM3)
```

e) Compruebe que la cola de transmisión se haya conmutado.

Compruebe si aparece el mensaje " AMQ7341 La cola de transmisión para el canal CL2.QM3 es XMITQ.CL2.QM3 " en el registro de errores del gestor de cola de pasarela.

Qué hacer a continuación

Pruebe la cola de transmisión separada enviando un mensaje de QM2 a Q1 en QM3 utilizando la definición de alias de cola Q1A

1. Ejecute el programa de ejemplo **amqsput** en QM2 para colocar un mensaje.

```
C:\IBM\MQ>amqsput Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A
```

```
Sample AMQSPUT0 end
```

2. Ejecute el programa de ejemplo **amqsget** para obtener el mensaje de Q1 en QM3

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

Conceptos relacionados

Control de accesos y varias colas de transmisión de clúster

Agrupación en clúster: Aislamiento de aplicaciones utilizando varias colas de transmisión de clúster

"Cómo trabajar con colas de transmisión de clúster y canales de clúster emisor" en la página 315

Los mensajes entre gestores de colas en clúster se almacenan en colas de transmisión de clúster y se reenvían por canales de clúster emisor. En cualquier momento, un canal de clúster emisor está asociado a una sola cola de transmisión. Si cambia la configuración del canal, éste puede cambiar a una cola de transmisión diferente la próxima vez que se inicie. El proceso de este conmutador está automatizado y es transaccional.

Tareas relacionadas

Añadir una definición de cola remota para aislar los mensajes enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza una definición de cola remota de clúster y un canal emisor y una cola de transmisión distintos.

Añadir un clúster y una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza un clúster adicional para aislar los mensajes a una cola de clúster determinada.

Modificar el valor predeterminado para separar colas de transmisión de clúster para aislar el tráfico de mensajes

Puede cambiar el modo predeterminado en que un gestor de colas almacena mensajes para una cola o un tema de clúster en una cola de transmisión. La modificación del valor predeterminado le permite aislar los mensajes de clúster en un gestor de colas de pasarela.

Agrupación en clúster: Planificación de cómo configurar las colas de transmisión de clúster

"Añadir un gestor de colas a un clúster: colas de transmisión separadas" en la página 337

Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando varias colas de transmisión de clúster.

Añadir un clúster y una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza un clúster adicional para aislar los mensajes a una cola de clúster determinada.

Antes de empezar

Los pasos de la tarea se graban para modificar la configuración ilustrada en la Figura 50 en la página 366.

1. El gestor de colas de pasarela debe estar en IBM MQ.

2. Construya los clústeres solapados que se muestran en Aplicación cliente-servidor desplegada en una arquitectura en estrella utilizando clústeres de IBM MQ en “Creación de dos clústeres solapados con un gestor de cola de pasarela” en la página 358 siguiendo los pasos de dicha tarea.
3. Siga los pasos de la Figura 50 en la página 366 del apartado “Añadir una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela” en la página 365 para crear la solución sin el clúster adicional. Utilice esto como base para los pasos de esta tarea.

Acerca de esta tarea

La solución para aislar el tráfico de mensajes a una sola aplicación en “Añadir una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela” en la página 365 funciona si la cola de clúster de destino es la única cola de clúster de un gestor de colas. Si no lo es, tiene dos opciones. Puede mover la cola a un gestor de colas diferente o crear un clúster que aísle la cola de otras colas de clúster en el gestor de colas.

Esta tarea le guía a través de los pasos necesarios para añadir un clúster para aislar la cola de destino. El clúster se añade sólo con este fin. En la práctica, recurra a la tarea de aislar ciertas aplicaciones de forma sistemática cuando se encuentre en el proceso de diseñar clústeres y esquemas de denominación de clúster. Añadir un clúster cada vez que una cola requiere aislamiento puede provocar que haya muchos clústeres para gestionar. En esta tarea, cambiará la configuración de “Añadir una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela” en la página 365 añadiendo un clúster `CL3` para aislar `Q1` en `QM3`. Las aplicaciones siguen ejecutándose durante el cambio.

Las definiciones nuevas y modificadas se resaltan en la Figura 51 en la página 370. El resumen de los cambios es el siguiente: Cree un clúster, lo que significa que también debe crear un nuevo clúster de repositorio completo. En el ejemplo, `QM3` se convierte en uno de los depósitos completos para `CL3`. Cree canales de clúster emisor y de clúster receptor para `QM1` para añadir el gestor de colas de pasarela al nuevo clúster. Cambie la definición de `Q1` para cambiarla a `CL3`. Modifique la lista de nombres de clúster en el gestor de colas de pasarela y añada una cola de transmisión de clúster para utilizar el nuevo canal de clúster. Por último, cambie el alias de cola `Q1A` a la nueva lista de nombres de clúster.

IBM MQ no puede transferir automáticamente mensajes de la cola de transmisión `XMITQ.CL2.QM3` que haya añadido en “Añadir una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela” en la página 365 a la nueva cola de transmisión `XMITQ.CL3.QM3`. Puede transferir mensajes automáticamente sólo si a ambas colas de transmisión les presta servicio el mismo canal de clúster emisor. En su lugar, la tarea describe una forma de realizar la conmutación manualmente, lo que puede ser adecuado para usted. Cuando la transferencia se ha completado, tiene la opción de volver a utilizar la cola de transmisión de clúster predeterminada para otras colas de clúster `CL2` en `QM3`. O puede seguir utilizando `XMITQ.CL2.QM3`. Si decide volver a una cola de transmisión de clúster predeterminada, el gestor de colas de pasarela gestiona la conmutación automáticamente.

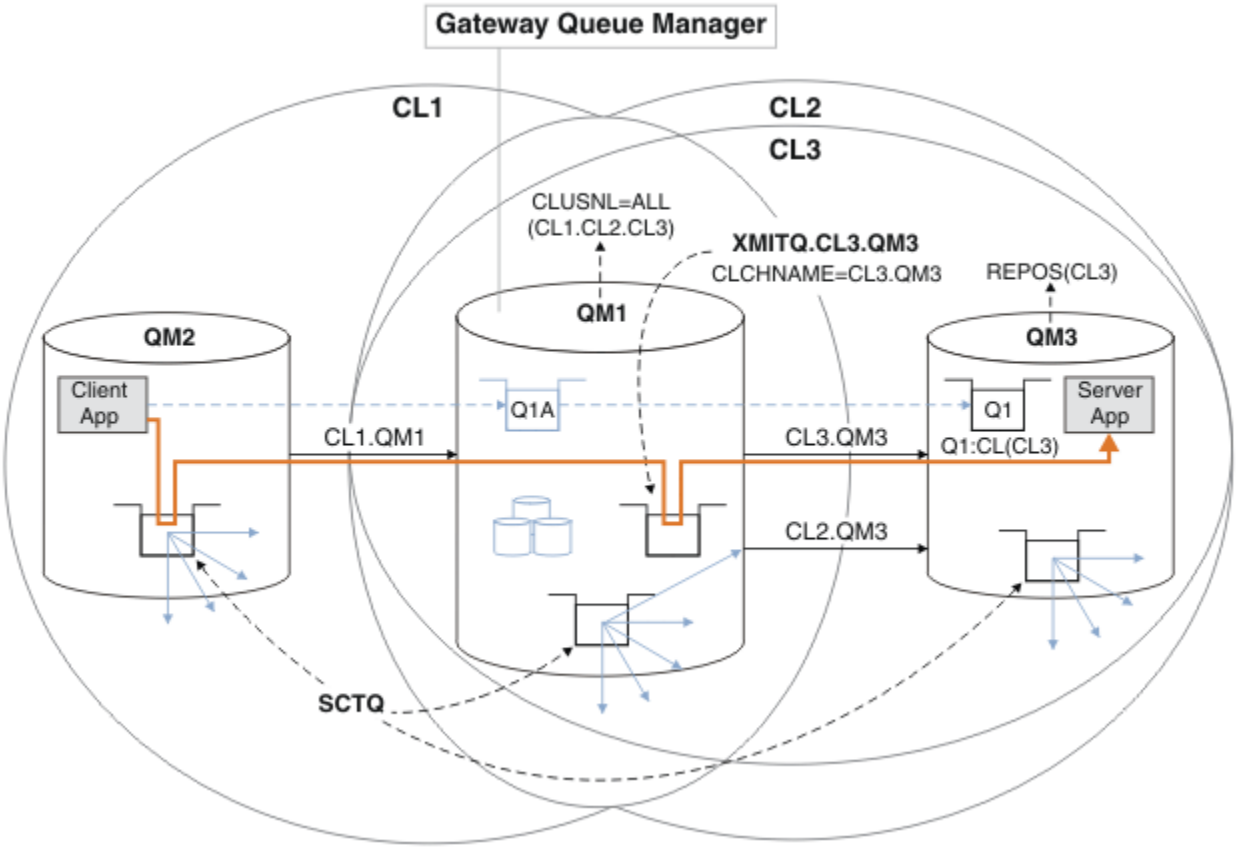


Figura 51. Uso de un clúster adicional para separar el tráfico de mensajes en el gestor de colas de pasarela dirigido a una de varias colas de clúster en el mismo gestor de colas

Procedimiento

1. Modificar los gestores de colas QM3 y QM5 para convertirlos en depósitos para CL2 y CL3.

   Para convertir un gestor de colas es un miembro de varios clústeres, debe utilizar una lista de nombres de clúster para identificar los clústeres de los que es miembro.

   ```
   *... On QM3 and QM5
   DEFINE NAMELIST(CL23) NAMES(CL2, CL3) REPLACE
   ALTER QMGR REPOS(' ') REPOSNL(CL23)
   ```

2. Defina los canales entre los gestores de colas QM3 y QM5 para CL3.

   ```
   *... On QM3
   DEFINE CHANNEL(CL3.QM5) CHLTYPE(CLUSSDR) CONNAME('localhost(1415)') CLUSTER(CL3) REPLACE
   DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSRCVR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE

   *... On QM5
   DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE
   DEFINE CHANNEL(CL3.QM5) CHLTYPE(CLUSRCVR) CONNAME('localhost(1415)') CLUSTER(CL3) REPLACE
   ```

3. Añada el gestor de colas de pasarela a CL3.

   Añada el gestor de colas de pasarela añadiendo QM1 a CL3 como depósito parcial. Cree un repositorio parcial añadiendo canales de clúster emisor y clúster receptor a QM1.

   Además, añada CL3 a la lista de nombres de todos los clústeres conectados al gestor de colas de pasarela.

   ```
   *... On QM1
   DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE
   ```

```
DEFINE CHANNEL(CL3.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL3) REPLACE
ALTER NAMELIST(ALL) NAMES(CL1, CL2, CL3)
```

4. Añada una cola de transmisión de clúster al gestor de colas de pasarela, QM1, para los mensajes que van a CL3 en QM3.

   Inicialmente, detenga la transferencia de mensajes del canal de clúster emisor desde la cola de transmisión hasta que esté preparado para conmutar las colas de transmisión.

   ```
   *... On QM1
   DEFINE QLOCAL(XMITQ.CL3.QM3) USAGE(XMITQ) CLCHNAME(CL3.QM3) GET(DISABLED) REPLACE
   ```

5. Drene los mensajes de la cola de transmisión de clúster existente XMITQ.CL2.QM3.

   Este subprocedimiento está destinado a preservar el orden de los mensajes en Q1 para que coincida con el orden en que llegaron al gestor de colas de pasarela. Con los clústeres, el orden de los mensajes no está totalmente garantizado, pero es probable. Si es necesario que el orden de los mensajes esté garantizado, las aplicaciones deben definir el orden de los mensajes; consulte Orden de recuperación de los mensajes de una cola.

   a) Cambie la cola de destino Q1 en QM3 de CL2 a CL3.

      ```
      *... On QM3
      ALTER QLOCAL(Q1) CLUSTER(CL3)
      ```

   b) Supervise XMITQ.CL3.QM3 hasta que se le empiecen a entregar mensajes.

      Los mensajes empiezan a ser entregados a XMITQ.CL3.QM3 cuando la conmutación de Q1 a CL3 se propaga al gestor de colas de pasarela.

      ```
      *... On QM1
      DISPLAY QUEUE(XMITQ.CL3.QM3) CURDEPTH
      ```

   c) Supervise XMITQ.CL2.QM3 hasta que no tenga mensajes en espera de ser entregados a Q1 en QM3.

      Nota: XMITQ.CL2.QM3 puede almacenar mensajes de otras colas en QM3 que son miembros de CL2, en cuyo caso la profundidad no puede ir a cero.

      ```
      *... On QM1
      DISPLAY QUEUE(XMITQ.CL2.QM3) CURDEPTH
      ```

   d) Habilite la obtención desde la nueva cola de transmisión de clúster, XMITQ.CL3.QM3

      ```
      *... On QM1
      ALTER QLOCAL(XMITQ.CL3.QM3) GET(ENABLED)
      ```

6. Elimine la cola de transmisión de clúster anterior, XMITQ.CL2.QM3, si ya no resulta necesaria.

   Los mensajes para las colas de clúster en CL2 en QM3 vuelven a utilizar la cola de transmisión de clúster predeterminada en el gestor de colas de pasarela, QM1. La cola de transmisión de clúster predeterminada es SYSTEM.CLUSTER.TRANSMIT.QUEUE o SYSTEM.CLUSTER.TRANSMIT.CL2.QM3. El hecho de que sea una o la otra depende de si el valor del atributo del gestor de colas **DEFCLXQ** en QM1 es SCTQ o CHANNEL. El gestor de colas transfiere mensajes desde XMITQ.CL2.QM3 automáticamente la próxima vez que el canal de clúster emisor CL2.QM3 se inicia.

   a) Cambie la cola de transmisión, XMITQ.CL2.QM3, para que deje de ser una cola de transmisión de clúster y se convierta en una cola de transmisión normal.

      De este modo se rompe la asociación de la cola de transmisión con cualquier canal de clúster emisor. En respuesta, IBM MQ transfiere automáticamente mensajes de XMITQ.CL2.QM3 a la cola de transmisión de clúster predeterminada cuando el canal de clúster emisor se inicia. Hasta entonces, los mensajes para CL2 en QM3 se siguen colocando en XMITQ.CL2.QM3.

```
*... On QM1
ALTER QLOCAL(XMITQ.CL2.QM3) CLCHNAME(' ')
```

b) Detenga el canal de clúster emisor CL2.QM3.

Al detener y reiniciar el canal de clúster emisor se inicia la transferencia de mensajes de XMITQ.CL2.QM3 a la cola de transmisión de clúster predeterminada. Normalmente, debería detener e iniciar el canal manualmente para iniciar la transferencia. La transferencia se inicia automáticamente si el canal se reinicia después de concluir al expirar su intervalo de desconexión.

```
*... On QM1
STOP CHANNEL(CL2.QM3)
```

La respuesta es que el mandato se ha aceptado:

```
AMQ8019: Stop IBM MQ channel accepted.
```

c) Compruebe que el canal CL2.QM3 se haya detenido

Si el canal no se detiene, puede volver a ejecutar el mandato **STOP CHANNEL** con la opción FORCE . Un ejemplo de definición de la opción FORCE sería si el canal no se detiene y no se puede reiniciar el otro gestor de colas para sincronizar el canal.

```
*... On QM1
DISPLAY CHSTATUS(CL2.QM3)
```

La respuesta es un resumen del estado de canal

```
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)                        CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))                CURRENT
RQMNAME(QM3)                            STATUS(STOPPED)
SUBSTATE(MQGET)                         XMITQ(XMITQ.CL2.QM3)
```

d) Inicie el canal, CL2.QM3.

```
*... On QM1
START CHANNEL(CL2.QM3)
```

La respuesta es que el mandato se ha aceptado:

```
AMQ8018: Start IBM MQ channel accepted.
```

e) Compruebe el canal iniciado.

```
*... On QM1
DISPLAY CHSTATUS(CL2.QM3)
```

La respuesta es un resumen del estado de canal:

```
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)      CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))  CURRENT
RQMNAME(QM3)        STATUS(RUNNING)
SUBSTATE(MQGET)      XMITQ(SYSTEM.CLUSTER.TRANSMIT. QUEUE|CL2.QM3)
```

f) Compruebe el registro de errores del gestor de colas de pasarela para ver si aparece el mensaje " AMQ7341 La cola de transmisión para el canal CL2.QM3 es SYSTEM.CLUSTER.TRANSMIT. *QUEUE|CL2.QM3* ".

g) Suprima la cola de transmisión de clúster, XMITQ.CL2.QM3.

```
*... On QM1
DELETE QLOCAL(XMITQ.CL2.QM3)
```

Qué hacer a continuación

Pruebe la cola en clúster separada enviando un mensaje de QM2 a Q1 en QM3 utilizando la definición de alias de cola Q1A

1. Ejecute el programa de ejemplo **amqsput** en QM2 para colocar un mensaje.

```
C:\IBM\MQ>amqsput Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A

Sample AMQSPUT0 end
```

2. Ejecute el programa de ejemplo **amqsget** para obtener el mensaje de Q1 en QM3

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

Conceptos relacionados

Control de accesos y varias colas de transmisión de clúster

Agrupación en clúster: Aislamiento de aplicaciones utilizando varias colas de transmisión de clúster

“Cómo trabajar con colas de transmisión de clúster y canales de clúster emisor” en la página 315

Los mensajes entre gestores de colas en clúster se almacenan en colas de transmisión de clúster y se reenvían por canales de clúster emisor. En cualquier momento, un canal de clúster emisor está asociado a una sola cola de transmisión. Si cambia la configuración del canal, éste puede cambiar a una cola de transmisión diferente la próxima vez que se inicie. El proceso de este conmutador está automatizado y es transaccional.

Tareas relacionadas

Añadir una definición de cola remota para aislar los mensajes enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza una definición de cola remota de clúster y un canal emisor y una cola de transmisión distintos.

Añadir una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza una cola de transmisión de clúster adicional para separar el tráfico de mensajes a un único gestor de colas de un clúster.

Modificar el valor predeterminado para separar colas de transmisión de clúster para aislar el tráfico de mensajes

Puede cambiar el modo predeterminado en que un gestor de colas almacena mensajes para una cola o un tema de clúster en una cola de transmisión. La modificación del valor predeterminado le permite aislar los mensajes de clúster en un gestor de colas de pasarela.

Agrupación en clúster: Planificación de cómo configurar las colas de transmisión de clúster

“Añadir un gestor de colas a un clúster: colas de transmisión separadas” en la página 337
Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando varias colas de transmisión de clúster.

Modificar el valor predeterminado para separar colas de transmisión de clúster para aislar el tráfico de mensajes

Puede cambiar el modo predeterminado en que un gestor de colas almacena mensajes para una cola o un tema de clúster en una cola de transmisión. La modificación del valor predeterminado le permite aislar los mensajes de clúster en un gestor de colas de pasarela.

Antes de empezar

1. El gestor de colas de pasarela debe estar en IBM MQ.
2. Construya los clústeres solapados que se muestran en Aplicación cliente-servidor desplegada en una arquitectura en estrella utilizando clústeres de IBM MQ en “Creación de dos clústeres solapados con un gestor de cola de pasarela” en la página 358 siguiendo los pasos de dicha tarea.

Acerca de esta tarea

Para implementar la arquitectura con varias colas de clústeres, el gestor de colas de pasarela debe estar en IBM MQ. Todo lo que debe hacer para utilizar varias colas de transmisión de clúster es cambiar el tipo de cola de transmisión de clúster predeterminado en el gestor de colas de pasarela. Cambie el valor del atributo del gestor de colas **DEFCLXQ** en `QM1` de `SCTQ` a `CHANNEL`; consulte Figura 52 en la página 375. El diagrama muestra un flujo de mensajes. Para los flujos a otros gestores de colas, o a otros clústeres, el gestor de colas crea colas de transmisión de clúster dinámicas permanentes adicionales. Cada canal de clúster emisor transfiere mensajes desde una cola de transmisión de clúster diferente.

El cambio no entra en vigor inmediatamente, a menos que conecte el gestor de colas de pasarela a clústeres por primera vez. La tarea incluye pasos para el típico caso de gestión de un cambio en una configuración existente. Para configurar un gestor de colas para utilizar colas de transmisión de clúster separadas cuando se une a un clúster por primera vez, consulte “Añadir un gestor de colas a un clúster: colas de transmisión separadas” en la página 337.

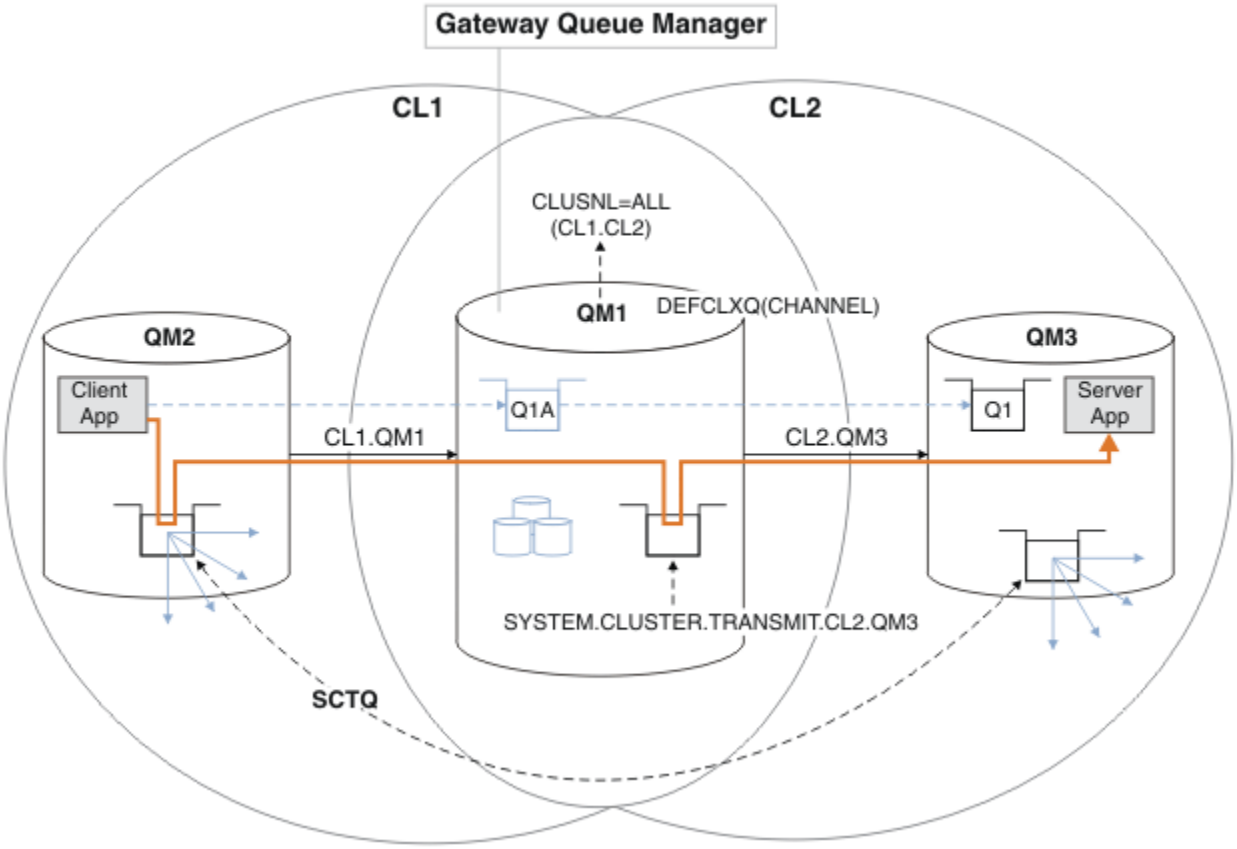


Figura 52. Aplicación cliente-servidor desplegada en una arquitectura en estrella con colas de transmisión de clúster distintas en el gestor de cola de pasarela.

Procedimiento

1. Cambie el gestor de colas de pasarela para utilizar colas de transmisión de clúster distintas.

```
*... On QM1
ALTER QMGR DEFCLXQ(CHANNEL)
```

2. Cambie a las colas de transmisión de clúster distintas.

   Cualquier canal de clúster emisor que no esté ejecutando conmutadores para utilizar colas de transmisión de clúster distintas la próxima vez que se inicie.

   Para conmutar los canales en ejecución, reinicie el gestor de colas o siga estos pasos:

   a) Liste los canales de clúster emisor que se ejecutan con `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
```

   La respuesta es una lista de informes de estado de canal:

```
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM2)            CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1412))    CURRENT
RQMNAME(QM2)                STATUS(RUNNING)
SUBSTATE(MQGET)             XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)            CHLTYPE(CLUSSDR)
```

```
CONNAME(127.0.0.1(1413))    CURRENT
RQMNAME(QM3)                STATUS(RUNNING)
SUBSTATE(MQGET)             XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM5)            CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1415))    CURRENT
RQMNAME(QM5)                STATUS(RUNNING)
SUBSTATE(MQGET)             XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM4)            CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1414))    CURRENT
RQMNAME(QM4)                STATUS(RUNNING)
SUBSTATE(MQGET)             XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

b) Detenga los canales que se están ejecutando

Para cada canal de la lista, ejecute el mandato:

```
*... On QM1
STOP CHANNEL(ChannelName)
```

Donde *ChannelName* es cada uno de `CL1.QM2, CL1.QM4, CL1.QM3, CL1.QM5`.

La respuesta es que el mandato se ha aceptado:

```
AMQ8019: Stop IBM MQ channel accepted.
```

c) Supervise qué canales se detienen

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
```

La respuesta es una lista de canales que todavía están en ejecución y canales que están detenidos:

```
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM2)           CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1412))   CURRENT
RQMNAME(QM2)               STATUS(STOPPED)
SUBSTATE( )                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)           CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))   CURRENT
RQMNAME(QM3)               STATUS(STOPPED)
SUBSTATE( )                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM5)           CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1415))   CURRENT
RQMNAME(QM5)               STATUS(STOPPED)
SUBSTATE( )                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM4)           CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1414))   CURRENT
RQMNAME(QM4)               STATUS(STOPPED)
SUBSTATE( )                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

d) Inicie cada canal detenido.

Realice este paso para todos los canales que se estaban ejecutando. Si un canal no se detiene, puede volver a ejecutar el mandato **STOP CHANNEL** con la opción FORCE . Un ejemplo de definición

de la opción FORCE sería si el canal no se detiene y no se puede reiniciar el otro gestor de colas para sincronizar el canal.

```
*... On QM1
START CHANNEL(CL2.QM5)
```

La respuesta es que el mandato se ha aceptado:

```
AMQ8018: Start IBM MQ channel accepted.
```

e) Supervise las colas de transmisión que se están conmutando.

Compruebe el registro de errores del gestor de colas de pasarela para ver si aparece el mensaje " AMQ7341 La cola de transmisión para el canal CL2.QM3 es SYSTEM.CLUSTER.TRANSMIT. *QUEUE*|*CL2.QM3* ".

f) Compruebe que SYSTEM.CLUSTER.TRANSMIT.QUEUE ya no se utilice

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
DISPLAY QUEUE(SYSTEM.CLUSTER.TRANSMIT.QUEUE) CURDEPTH
```

La respuesta es una lista de informes de estado de canal y la profundidad de SYSTEM.CLUSTER.TRANSMIT.QUEUE:

```
AMQ8420: Channel Status not found.
AMQ8409: Display Queue details.
QUEUE(SYSTEM.CLUSTER.TRANSMIT.QUEUE)    TYPE(QLOCAL)
CURDEPTH(0)
```

g) Supervise qué canales se inician

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
```

La respuesta es una lista de los canales, en este caso que ya se están ejecutando con las nuevas colas de transmisión de clúster predeterminadas:

```
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM2)                        CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1412))                CURRENT
RQMNAME(QM2)                            STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL1.QM2)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)                        CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))                CURRENT
RQMNAME(QM3)                            STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL2.QM3)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM5)                        CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1415))                CURRENT
RQMNAME(QM5)                            STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL2.QM5)
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM4)                        CHLTYPE(CLUSSDR)
```

```
CONNAME(127.0.0.1(1414))                CURRENT
RQMNAME(QM4)                            STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL1.QM4)
```

Qué hacer a continuación

1. Pruebe la cola de transmisión definida automáticamente enviando un mensaje de QM2 a Q1 en QM3, resolviendo el nombre de cola con la definición de alias de cola Q1A
   a. Ejecute el programa de ejemplo **amqsput** en QM2 para colocar un mensaje.

```
C:\IBM\MQ>amqsput Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A

Sample AMQSPUT0 end
```

   b. Ejecute el programa de ejemplo **amqsget** para obtener el mensaje de Q1 en QM3

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

2. Considere si se debe volver a configurar la seguridad, configurando la seguridad para las colas de clúster en los gestores de colas en los que se originan los mensajes para las colas de clúster.

Conceptos relacionados

Control de accesos y varias colas de transmisión de clúster

Agrupación en clúster: Aislamiento de aplicaciones utilizando varias colas de transmisión de clúster

Tareas relacionadas

Añadir una definición de cola remota para aislar los mensajes enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza una definición de cola remota de clúster y un canal emisor y una cola de transmisión distintos.

Añadir una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza una cola de transmisión de clúster adicional para separar el tráfico de mensajes a un único gestor de colas de un clúster.

Añadir un clúster y una cola de transmisión de clúster para aislar el tráfico de mensajes de clúster enviados desde un gestor de colas de pasarela

Modifique la configuración de clústeres solapados que utilizan un gestor de colas de pasarela. Tras la modificación se transfieren mensajes a una aplicación desde el gestor de colas de pasarela sin utilizar la misma cola de transmisión o canales como otros mensajes de clúster. La solución utiliza un clúster adicional para aislar los mensajes a una cola de clúster determinada.

Agrupación en clúster: Planificación de cómo configurar las colas de transmisión de clúster

“Añadir un gestor de colas a un clúster: colas de transmisión separadas” en la página 337

Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando varias colas de transmisión de clúster.

Eliminar una cola de clúster de un gestor de colas

Inhabilite la cola INVENTQ en Toronto. Envíe todos los mensajes de inventario a Nueva York, y suprima la cola INVENTQ en Toronto cuando esté vacía.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- El clúster de INVENTORY se ha configurado como se describe en "Añadir un gestor de colas que aloja una cola" en la página 342. Contiene cuatro gestores de colas. LONDON y NEWYORK contienen ambos repositorios completos. PARIS y TORONTO contienen repositorios parciales. La aplicación de inventario se ejecuta en los sistemas de Nueva York y Toronto y se activa con la llegada de mensajes a la cola INVENTQ.
- Debido a que la carga de trabajo se ha reducido, ya no desea ejecutar la aplicación de inventario en Toronto. Desea inhabilitar la cola INVENTQ alojada por el gestor de colas TORONTO, y que TORONTO suministre mensajes a la cola INVENTQ en NEWYORK.
- Existe conectividad de red entre los cuatro sistemas.
- El protocolo de red es TCP.

Acerca de esta tarea

Siga estos pasos para eliminar una cola de clúster.

Procedimiento

1. Indique que la cola ya no está disponible.

   Para eliminar una cola de un clúster, elimine el nombre de clúster de la definición de cola local. Modifique INVENTQ en TORONTO para que no sea accesible desde el resto del clúster:

   ```
   ALTER QLOCAL(INVENTQ) CLUSTER(' ')
   ```

2. Compruebe que la cola ya no está disponible.

   En un gestor de colas de repositorio completo, ya sea LONDON o NEWYORK, compruebe que la cola ya no está alojada por el gestor de colas TORONTO emitiendo el mandato siguiente:

   ```
   DIS QCLUSTER (INVENTQ)
   ```

   TORONTO no aparece en los resultados, si el mandato ALTER se ha completado satisfactoriamente.

3. Inhabilite la cola.

   Inhabilite la cola INVENTQ en TORONTO para que no se puedan grabar más mensajes en ella:

   ```
   ALTER QLOCAL(INVENTQ) PUT(DISABLED)
   ```

   Ahora, los mensajes en tránsito hacia esta cola que utilizan MQOO_BIND_ON_OPEN van a la cola de mensajes no entregados. Debe hacer que todas las aplicaciones dejen de transferir mensajes explícitamente a la cola en este gestor de colas.

4. Supervise la cola hasta que esté vacía.

Supervise la cola mediante el mandato DISPLAY QUEUE, especificando los atributos IPPROCS, OPPROCS y CURDEPTH, o utilice el mandato **WRKMQMSTS** en IBM i. Cuando el número de procesos de entrada y de salida y la profundidad actual de la cola son todos cero, la cola está vacía.

5. Supervise el canal para asegurarse de que no hay mensajes pendientes.

   Para asegurarse de que no hay mensajes pendientes en el canal INVENTORY.TORONTO, supervise el canal de clúster emisor llamado INVENTORY.TORONTO en cada uno de los otros gestores de colas. Emita el mandato DISPLAY CHSTATUS especificando el parámetro INDOUBT desde cada gestor de colas:

   ```
   DISPLAY CHSTATUS(INVENTORY.TORONTO) INDOUBT
   ```

   Si hay algún mensaje pendiente, debe resolverlo antes de continuar. Por ejemplo, puede probar a emitir el mandato RESOLVE CHANNEL o a detener y reiniciar el canal.

6. Suprima la cola local.

   Cuando esté convencido de que no hay más mensajes para entregar a la aplicación de inventario en TORONTO, puede suprimir la cola:

   ```
   DELETE QLOCAL(INVENTQ)
   ```

7. Ahora puede eliminar la aplicación de inventario del sistema en Toronto

   Eliminar la aplicación evita la duplicación y ahorra espacio en el sistema.

Resultados

El clúster configurado por esta tarea es como el configurado por la tarea anterior. La diferencia es que la cola INVENTQ ya no está disponible en el gestor de colas TORONTO.

Cuando dejó la cola fuera de servicio en el paso 1, el gestor de colas TORONTO envió un mensaje a los dos gestores de colas de repositorio completo. Les informó del cambio en el estado. Los gestores de colas de repositorio completo transmiten esta información a otros gestores de colas del clúster que han solicitado actualizaciones de la información relativa a la cola INVENTQ.

Cuando un gestor de colas coloca un mensaje en la cola INVENTQ, el repositorio parcial actualizado indica que la cola INVENTQ sólo está disponible en el gestor de colas NEWYORK. El mensaje se envía al gestor de colas NEWYORK.

Qué hacer a continuación

En esta tarea, había sólo una cola para eliminar y sólo un clúster del que eliminarla.

Supongamos que hay muchas colas que hacen referencia a una lista de nombres que contiene muchos nombres de clúster. Por ejemplo, el gestor de colas TORONTO podría alojar no sólo la cola INVENTQ, sino también las colas PAYROLLQ, SALESQ y PURCHASESQ. TORONTO pone estas colas a disposición de todos los clústeres apropiados, INVENTORY, PAYROLL, SALES y PURCHASES. Defina una lista de los nombres de clúster en el gestor de colas TORONTO:

```
DEFINE NAMELIST(TOROLIST)
DESCR('List of clusters TORONTO is in')
NAMES(INVENTORY, PAYROLL, SALES, PURCHASES)
```

Añada la lista de nombres a cada definición de cola:

```
DEFINE QLOCAL(INVENTQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(PAYROLLQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(SALESQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(PURCHASESQ) CLUSNL(TOROLIST)
```

Ahora supongamos que desea eliminar todas esas colas del clúster SALES, porque la operación SALES va a ser reemplazada por la operación PURCHASES. Lo único que tiene que hacer es modificar la lista de nombres TOROLIST para eliminar el nombre del clúster SALES de la misma.

Si desea eliminar una sola cola de uno de los clústeres en la lista de nombres, cree una lista de nombres que contenga la lista de nombres de clúster restantes. A continuación, modifique la definición de cola para utilizar la nueva lista de nombres. Para eliminar la cola PAYROLLQ del clúster INVENTORY:

1. Cree una lista de nombres:

```
DEFINE NAMELIST(TOROSHORTLIST)
DESCR('List of clusters TORONTO is in other than INVENTORY')
NAMES(PAYROLL, SALES, PURCHASES)
```

2. Modifique la definición de cola PAYROLLQ:

```
ALTER QLOCAL(PAYROLLQ) CLUSNL(TOROSHORTLIST)
```

Eliminación de un gestor de colas de un clúster: práctica recomendada

Elimine un gestor de colas de un clúster en los casos en los que el gestor de colas pueda comunicarse con normalidad con al menos un repositorio completo del clúster.

Antes de empezar

Este método es el recomendado para los casos en los que al menos hay un repositorio completo disponible con el que puede contactar el gestor de colas que se está eliminando. Este método requiere una intervención manual mínima y permite que el gestor de colas negocie una retirada controlada del clúster. Si el gestor de colas que se está eliminando no puede contactar con un repositorio completo, consulte “Eliminación de un gestor de colas de un clúster: método alternativo” en la página 383.

Acerca de esta tarea

Esta tarea de ejemplo elimina el gestor de colas LONDON del clúster INVENTORY. El clúster INVENTORY se ha configurado como se describe en “Añadir un gestor de colas a un clúster” en la página 335 y se ha modificado como se describe en “Eliminar una cola de clúster de un gestor de colas” en la página 379.

El proceso de eliminar un gestor de colas de un clúster es más complejo que el proceso de añadir un gestor de colas.

Cuando un gestor de colas se une a un clúster, los miembros existentes del clúster no tienen conocimiento sobre el nuevo gestor de colas y, por lo tanto, no interactúan con él. Deben crearse nuevos canales de emisor y receptor en el gestor de colas que se une de modo que pueda conectarse a un repositorio completo.

Cuando se elimina un gestor de colas de un clúster, es muy probable que aplicaciones conectadas al gestor de colas utilicen objetos, tales como colas, alojados en algún otro lugar del clúster. Asimismo, las aplicaciones que están conectadas a otros gestores de colas del clúster pueden estar utilizando objeto alojados en el gestor de colas de destino. Como resultado de estas aplicaciones, el gestor de colas actual puede crear canales emisor adicionales para establecer comunicación con otros miembros del clúster que no sean el repositorio completo que se utiliza para unirse al clúster. Cada uno de los gestores de colas del clúster tiene una copia en la memoria caché de datos que describe a otros miembros del clúster. Puede incluir el que se está eliminando.

Procedimiento

1. Antes de eliminar el gestor de colas del clúster, asegúrese de que el gestor de colas ya no aloja recursos que necesita el clúster:
   - Si el gestor de colas aloja un repositorio completo, realice los pasos 1 a 6 de la tarea “Trasladar un depósito completo a otro gestor de colas” en la página 347. Si la funcionalidad de repositorio

completo del gestor de colas que se va a eliminar no se trasladará a otro gestor de colas, sólo es necesario realizar los pasos 5 y 6.

- Si el gestor de colas aloja colas de clúster, realice los pasos 1 a 7 de la tarea "Eliminar una cola de clúster de un gestor de colas" en la página 379.
- Si el gestor de colas aloja temas de clúster, suprima los temas (por ejemplo, mediante el mandato DELETE TOPIC) o traspáselas a otros hosts como se describe en "Mover una definición de tema de clúster a un gestor de colas diferente" en la página 463.

  Nota: Si elimina un gestor de colas del clúster y el gestor de colas todavía aloja un tema de clúster, es posible que el gestor de colas continúe intentando entregar publicaciones a los gestores de colas que quedan en el clúster hasta que se suprima el tema.

2. Modifique los canales de recepción del clúster definidos manualmente para eliminarlos del clúster, en el gestor de colas LONDON:

```
ALTER CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) CLUSTER(' ')
```

3. Modifique los canales de clúster emisor definidos manualmente para eliminarlos del clúster, en el gestor de colas LONDON:

```
ALTER CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSSDR) CLUSTER(' ')
```

   Los otros gestores de colas del clúster aprenden que este gestor de colas y sus recursos de clúster ya no forman parte del clúster.

4. Supervise la cola de transmisión del clúster, en el gestor de colas LONDON, hasta que no quede ningún mensaje a la espera de ser transmitido a ningún repositorio completo que quede en el clúster.

```
DISPLAY CHSTATUS(INVENTORY.PARIS) XQMSGSA
```

   Si quedan mensajes en la cola de transmisión, determine porqué no se envían a los repositorios completos PARIS y NEWYORK antes de continuar.

Resultados

El gestor de colas LONDON ya no forma parte del clúster. No obstante, puede seguir funcionando como gestor de colas independiente.

Qué hacer a continuación

El resultado de estos cambios puede confirmarse en los miembros restantes del clúster emitiendo el siguiente mandato:

```
DISPLAY CLUSQMGR(LONDON)
```

El gestor de colas sigue visualizándose hasta que se hayan detenido los canales de clúster emisor definidos automáticamente. Puede esperar a que esto suceda o continuar supervisando las instancias activas emitiendo el siguiente mandato:

```
DISPLAY CHANNEL(INVENTORY.LONDON)
```

Si está seguro de que no se están entregando más mensajes a este gestor de colas, puede detener los canales de clúster emisor en LONDON emitiendo el siguiente mandato en los miembros restantes del clúster:

```
STOP CHANNEL(INVENTORY.LONDON) STATUS(INACTIVE)
```

Una vez propagados los cambios por todo el clúster, si ya no se entregan más mensajes a este gestor de colas, detenga y suprima el canal CLUSRCVR en LONDON:

```
STOP CHANNEL(INVENTORY.LONDON)
DELETE CHANNEL(INVENTORY.LONDON)
```

Si se estaba utilizando una cola de transmisión definida manualmente para este canal, y el patrón CLCHNAME no coincide con ningún otro canal existente o planificado, es posible que desee suprimir la cola de transmisión. Por ejemplo:

```
DELETE QLOCAL(PARIS.CUSTOM.XMITQ)
```

Nota: Si las colas de transmisión definidas automáticamente o el SYSTEM.CLUSTER.TRANSMIT.QUEUE están en uso, este paso no es necesario.

El gestor de colas eliminado puede volver a añadirse al clúster más adelante como se describe en “Añadir un gestor de colas a un clúster” en la página 335. El gestor de colas eliminado continúa almacenando en la memoria caché información de los miembros restantes del clúster durante un máximo de 90 días. Si prefiere no esperar hasta que esta memoria caché caduque, puede forzar su eliminación como se describe en “Restauración de un gestor de colas al estado previo al clúster” en la página 385.

Tareas relacionadas

Eliminación de un gestor de colas de un clúster (utilizando IBM MQ Explorer)

Referencia relacionada

ALTER CHANNEL (alterar valores de canal)

DISPLAY CHANNEL (visualizar definición de canal)

DISPLAY CHSTATUS (visualizar estado de canal)

DISPLAY CLUSQMGR (visualizar información de canal para gestores de colas de clúster)

STOP CHANNEL (detener un canal)

Eliminación de un gestor de colas de un clúster: método alternativo

Elimine un gestor de colas de un clúster en los casos en los que, debido a un problema importante del sistema o de la configuración, el gestor de colas no pueda comunicarse con normalidad con ningún repositorio completo del clúster.

Antes de empezar

Este método alternativo de eliminar un gestor de colas de un clúster detiene manualmente un clúster y detiene todos los canales del clúster que enlazan el gestor de colas con el clúster, eliminando forzosamente el gestor de colas del clúster. Este método se utiliza en los casos en los que el gestor de colas que se está eliminado no se puede comunicar con ningún repositorio completo. Esto puede ser debido, por ejemplo, a que el gestor de colas ha dejado de funcionar o a que se ha producido un error de comunicaciones prolongado entre el gestor de colas y el clúster. De lo contrario, utilice el método más común: “Eliminación de un gestor de colas de un clúster: práctica recomendada” en la página 381.

Acerca de esta tarea

Esta tarea de ejemplo elimina el gestor de colas LONDON del clúster INVENTORY. El clúster INVENTORY se ha configurado como se describe en “Añadir un gestor de colas a un clúster” en la página 335 y se ha modificado como se describe en “Eliminar una cola de clúster de un gestor de colas” en la página 379.

El proceso de eliminar un gestor de colas de un clúster es más complejo que el proceso de añadir un gestor de colas.

Cuando un gestor de colas se une a un clúster, los miembros existentes del clúster no tienen conocimiento sobre el nuevo gestor de colas y, por lo tanto, no interactúan con él. Deben crearse nuevos canales de emisor y receptor en el gestor de colas que se une de modo que pueda conectarse a un repositorio completo.

Cuando se elimina un gestor de colas de un clúster, es muy probable que aplicaciones conectadas al gestor de colas utilicen objetos, tales como colas, alojados en algún otro lugar del clúster. Asimismo, las aplicaciones que están conectadas a otros gestores de colas del clúster pueden estar utilizando objeto alojados en el gestor de colas de destino. Como resultado de estas aplicaciones, el gestor de colas actual puede crear canales emisor adicionales para establecer comunicación con otros miembros del clúster que no sean el repositorio completo que se utiliza para unirse al clúster. Cada uno de los gestores de colas del clúster tiene una copia en la memoria caché de datos que describe a otros miembros del clúster. Puede incluir el que se está eliminando.

Este procedimiento puede ser apropiado en un caso de emergencia, cuando no es posible esperar a que el gestor de colas abandone el clúster correctamente.

Procedimiento

1. Antes de eliminar el gestor de colas del clúster, asegúrese de que el gestor de colas ya no aloja recursos que necesita el clúster:
   - Si el gestor de colas aloja un repositorio completo, realice los pasos 1 a 6 de la tarea "Trasladar un depósito completo a otro gestor de colas" en la página 347. Si la funcionalidad de repositorio completo del gestor de colas que se va a eliminar no se trasladará a otro gestor de colas, sólo es necesario realizar los pasos 5 y 6.
   - Si el gestor de colas aloja colas de clúster, realice los pasos 1 a 7 de la tarea "Eliminar una cola de clúster de un gestor de colas" en la página 379.
   - Si el gestor de colas aloja temas de clúster, suprima los temas (por ejemplo, mediante el mandato DELETE TOPIC) o traspáselas a otros hosts como se describe en "Mover una definición de tema de clúster a un gestor de colas diferente" en la página 463.

     Nota: Si elimina un gestor de colas del clúster y el gestor de colas todavía aloja un tema de clúster, es posible que el gestor de colas continúe intentando entregar publicaciones a los gestores de colas que quedan en el clúster hasta que se suprima el tema.
2. Detenga todos los canales utilizados para establecer comunicación con otros gestores de colas del clúster. Utilice `MODE(FORCE)` para detener el canal `CLUSRCVR` en el gestor de colas `LONDON`. De lo contrario, es posible que tenga que esperar a que el gestor de colas emisor detenga el canal:

   ```
   STOP CHANNEL(INVENTORY.LONDON) MODE(FORCE)
   STOP CHANNEL(INVENTORY.TORONTO)
   STOP CHANNEL(INVENTORY.PARIS)
   STOP CHANNEL(INVENTORY.NEWYORK)
   ```
3. Supervise los estados de los canales del gestor de colas `LONDON`, hasta que se detengan los canales:

   ```
   DISPLAY CHSTATUS(INVENTORY.LONDON)
   DISPLAY CHSTATUS(INVENTORY.TORONTO)
   DISPLAY CHSTATUS(INVENTORY.PARIS)
   DISPLAY CHSTATUS(INVENTORY.NEWYORK)
   ```

   No se envían más mensajes de aplicación hacia ni desde otros gestores de aplicaciones del clúster una vez se han detenido los canales.
4. Suprima los canales del clúster definidos manualmente en el gestor de colas `LONDON`:

   ```
   DELETE CHANNEL(INVENTORY.NEWYORK)
   DELETE CHANNEL(INVENTORY.TORONTO)
   ```

5. El resto de gestores de colas del clúster sigue conservando la información sobre el gestor de colas eliminado y puede continuar enviándole mensajes. Para depurar la información de los restantes gestores de colas, restaure el gestor de colas eliminado desde el clúster en uno de los repositorios completos:

   ```
   RESET CLUSTER(INVENTORY) ACTION(FORCEREMOVE) QMNAME(LONDON) QUEUES(YES)
   ```

   Si puede haber otro gestor de colas en el clúster que tenga el mismo nombre que el gestor de colas eliminado, especifique el **QMID** del gestor de colas eliminado.

Resultados

El gestor de colas LONDON ya no forma parte del clúster. No obstante, puede seguir funcionando como gestor de colas independiente.

Qué hacer a continuación

El resultado de estos cambios puede confirmarse en los miembros restantes del clúster emitiendo el siguiente mandato:

```
DISPLAY CLUSQMGR(LONDON)
```

El gestor de colas sigue visualizándose hasta que se hayan detenido los canales de clúster emisor definidos automáticamente. Puede esperar a que esto suceda o continuar supervisando las instancias activas emitiendo el siguiente mandato:

```
DISPLAY CHANNEL(INVENTORY.LONDON)
```

Una vez propagados los cambios por todo el clúster, si ya no se entregan más mensajes a este gestor de colas, suprima el canal CLUSRCVR en LONDON:

```
DELETE CHANNEL(INVENTORY.LONDON)
```

El gestor de colas eliminado puede volver a añadirse al clúster más adelante como se describe en “Añadir un gestor de colas a un clúster” en la página 335. El gestor de colas eliminado continúa almacenando en la memoria caché información de los miembros restantes del clúster durante un máximo de 90 días. Si prefiere no esperar hasta que esta memoria caché caduque, puede forzar su eliminación como se describe en “Restauración de un gestor de colas al estado previo al clúster” en la página 385.

Referencia relacionada

DELETE CHANNEL (suprimir un canal)
DISPLAY CHANNEL (visualizar definición de canal)
DISPLAY CHSTATUS (visualizar estado de canal)
DISPLAY CLUSQMGR (visualizar información de canal para gestores de colas de clúster)
STOP CHANNEL (detener un canal)
RESET CLUSTER (restablecer un clúster)

Restauración de un gestor de colas al estado previo al clúster

Cuando se elimina un gestor de colas de un clúster, mantiene información de los miembros del clúster restantes. Finalmente, esta información caduca y se suprime automáticamente. No obstante, si prefiere suprimirla inmediatamente, puede seguir los pasos que se indican en este tema.

Antes de empezar

Se presupone que el gestor de colas que se ha eliminado del clúster ya no lleva a cabo ningún trabajo en el clúster. Por ejemplo, sus colas ya no reciben mensajes del clúster y no hay ninguna aplicación a la espera de que lleguen mensajes en estas colas.

Acerca de esta tarea

Cuando se elimina un gestor de colas de un clúster, mantiene información de los miembros del clúster restantes durante un máximo de 90 días. Esto puede tener ventajas para el sistema, en especial si el gestor de colas vuelve a unirse al clúster rápidamente. Cuando caduca esta información, se suprime automáticamente. Sin embargo, es posible que prefiera suprimir esta información manualmente por algún motivo. Por ejemplo:

- Es posible que desee confirmar que ha detenido cada una de las aplicaciones de este gestor de colas que anteriormente utilizaban los recursos del clúster. Hasta que no caduca la información de los miembros del clúster restantes, estas aplicaciones continúan grabando en una cola de transmisión. Cuando se suprime la información de clúster, el sistema genera un error si una aplicación de este tipo intenta utilizar los recursos del clúster.
- Cuando visualiza la información de estado para el gestor de colas, es posible que prefiera no ver la información acerca de los restantes miembros del clúster que está caducando.

Esta tarea utiliza como ejemplo el clúster INVENTORY. Se ha eliminado el gestor de colas LONDON del clúster INVENTORY, como se describe en "Eliminación de un gestor de colas de un clúster: práctica recomendada" en la página 381. Para suprimir la información del resto de los miembros del clúster, emita los siguientes mandatos en el gestor de colas LONDON.

Procedimiento

1. Elimine toda la memoria de los demás gestores de colas del clúster de este gestor de colas:

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

2. Supervise el gestor de colas hasta que todos los recursos del clúster hayan desaparecido:

```
DISPLAY CLUSQMGR(*) CLUSTER(INVENTORY)
DISPLAY QCLUSTER(*) CLUSTER(INVENTORY)
DISPLAY TOPIC(*) CLUSTER(INVENTORY)
```

Conceptos relacionados

Clústeres

Componentes de clúster

Referencia relacionada

Comparación de agrupación en clúster y gestión de colas distribuidas

Realizar el mantenimiento de un gestor de colas

Suspenda y reanude un gestor de colas de un clúster para realizar el mantenimiento.

Acerca de esta tarea

De vez en cuando, es posible que tenga que realizar tareas de mantenimiento en un gestor de colas que forma parte de un clúster. Por ejemplo, puede que tenga que realizar copias de seguridad de los datos de sus colas, o aplicar arreglos en el software. Si el gestor de colas aloja alguna cola, deben suspenderse sus actividades. Cuando el mantenimiento se haya completado, sus actividades se pueden reanudar.

Procedimiento

1. Suspenda un gestor de colas emitiendo el mandato **runmqsc** SUSPEND QMGR:

   ```
   SUSPEND QMGR CLUSTER(SALES)
   ```

   El mandato **runmqsc** SUSPEND informa a los gestores de colas en el clúster SALES que este gestor de colas se ha suspendido.

   La finalidad del mandato SUSPEND QMGR es únicamente advertir a otros gestores de colas que eviten enviar mensajes a este gestor de colas si es posible. No significa que el gestor de colas esté inhabilitado. Se le siguen enviando algunos mensajes que tienen que ser manejados por este gestor de colas, por ejemplo cuando este gestor de colas es el único host de una cola en clúster.

   Mientras el gestor de colas está suspendido, las rutinas de gestión de carga de trabajo evitan enviarle mensajes. Los mensajes que tienen que ser manejados por ese gestor de colas incluyen los mensajes enviados por el gestor de colas local.

   IBM MQ utiliza un algoritmo de equilibrio de carga de trabajo para determinar qué destinos son adecuados, en lugar de seleccionar el gestor de colas local siempre que sea posible.

   a) Fuerce la suspensión de un gestor de colas utilizando la opción FORCE en el mandato SUSPEND QMGR:

      ```
      SUSPEND QMGR CLUSTER(SALES) MODE(FORCE)
      ```

      MODE(FORCE) detiene forzadamente todos los canales de entrada de otros gestores de colas del clúster. Si no especifica MODE(FORCE), se aplica el valor predeterminado MODE(QUIESCE).
2. Realice todas las tareas de mantenimiento que sean necesarias.
3. Reanude el gestor de colas emitiendo el mandato **runmqsc** RESUME QMGR:

   ```
   RESUME QMGR CLUSTER(SALES)
   ```

Resultados

El mandato **runmqsc** RESUME informa a los repositorios completos que el gestor de colas está de nuevo disponible. Los gestores de colas de repositorio completo difunden esta información a otros gestores de colas que han solicitado actualizaciones de información relativa a este gestor de colas.

Realizar el mantenimiento de la cola de transmisión de clúster

Haga un esfuerzo por mantener disponibles las colas de transmisión de clúster. Son esenciales para el rendimiento de los clústeres. z/OS En z/OS, establezca INDXTYPE de una cola de transmisión de clúster en CORRELID.

Antes de empezar

- Asegúrese de que la cola de transmisión de clúster no se llene.
- Procure no emitir accidentalmente un mandato **runmqsc** ALTER para establecerla en get disabled o put disabled.
- Asegúrese de que el soporte en el que se almacena la cola de transmisión de clúster en la página z/OS (por ejemplo conjuntos de páginas z/OS) no se llene.

Acerca de esta tarea

z/OS

El siguiente procedimiento sólo es aplicable a z/OS.

Procedimiento

Establezca el INDXTYPE de la cola de transmisión de clúster en CORRELID

Renovar un gestor de colas de clúster

Puede eliminar canales definidos automáticamente y objetos de clúster definidos automáticamente del repositorio local utilizando el mandato REFRESH CLUSTER . No se pierde ningún mensaje.

Antes de empezar

Es posible que el Centro de soporte de IBM le pida que utilice el mandato. No utilice el mandato sin una cuidadosa consideración. Por ejemplo, para clústeres grandes, el uso del mandato **REFRESH CLUSTER** puede ser perjudicial para el clúster mientras está en curso, y de nuevo a intervalos de 27 días a partir de entonces, cuando los objetos de clúster envían automáticamente actualizaciones de estado a todos los gestores de colas interesados. Consulte Agrupación en clúster: Uso de procedimientos recomendados de REFRESH CLUSTER.

Acerca de esta tarea

Un gestor de colas puede empezar desde cero en un clúster. En circunstancias normales, no necesita utilizar el mandato REFRESH CLUSTER.

Procedimiento

Emita el mandato REFRESH CLUSTER **MQSC** desde un gestor de colas para eliminar el gestor de colas de clúster definido automáticamente y los objetos de cola del repositorio local.

El mandato sólo elimina objetos que hacen referencia a otros gestores de colas, no elimina objetos relacionados con el gestor de colas local. El mandato también elimina canales definidos automáticamente. Elimina canales que no tienen mensajes en la cola de transmisión de clúster y que no están conectados a un gestor de colas de repositorio completo.

Resultados

En efecto, el mandato REFRESH CLUSTER permite que un gestor de colas se arranque en frío con respecto al contenido de su repositorio completo. IBM MQ se asegura de que no se pierden datos de las colas.

Información relacionada

Agrupación en clúster: utilización de las recomendaciones de REFRESH CLUSTER

Recuperación de un gestor de colas de clúster

Actualice la información del clúster sobre un gestor de colas mediante el mandato **runmqsc** REFRESH CLUSTER. Siga este procedimiento después de recuperar un gestor de colas desde una copia de seguridad de punto en el tiempo.

Antes de empezar

Ha restaurado un gestor de colas de clúster a partir de una copia de seguridad puntual.

Acerca de esta tarea

Para recuperar un gestor de colas de un clúster, restaure el gestor de colas y luego actualice la información de clúster mediante el mandato **runmqsc** REFRESH CLUSTER.

Nota: Para los clústeres de gran tamaño, el uso del mandato **REFRESH CLUSTER** puede interrumpir el clúster mientras está en curso y, a partir de entonces, de nuevo a intervalos de 27 días cuando los objetos de clúster envían automáticamente actualizaciones de estado a todos los gestores de colas interesados. Consulte La renovación en un clúster grande puede afectar el rendimiento y la disponibilidad del clúster.

Procedimiento

Emita el mandato REFRESH CLUSTER en el gestor de colas restaurado para todos los clústeres en los que el gestor de colas participa.

Qué hacer a continuación

No hace falta emitir el mandato REFRESH CLUSTER en ningún otro gestor de colas.

Conceptos relacionados

Agrupación en clúster: utilización de las recomendaciones de REFRESH CLUSTER

Configurar canales de clúster para disponibilidad

Siga buenas prácticas de configuración para mantener los canales de clúster ejecutándose sin contratiempos si hay paros de red intermitentes.

Antes de empezar

Los clústeres le eximen de la necesidad de definir canales, pero sigue teniendo que realizar su mantenimiento. Se utiliza la misma tecnología de canal para la comunicación entre los gestores de colas de un clúster que la que se utiliza en la gestión de colas distribuidas. Para entender los canales de clúster, debe estar familiarizado con temas tales como:

- Cómo funcionan los canales
- Cómo averiguar su estado
- Cómo utilizar las salidas de canal

Acerca de esta tarea

Es posible que desee prestar especial atención a los puntos siguientes:

Procedimiento

Tenga en cuenta los siguientes puntos cuando configure canales de clúster

- Elija valores para HBINT o KAINT en canales de clúster emisor y canales de clúster receptor que no sobrecarguen la red con muchos flujos de pulsaciones o de mantener activo. Un intervalo inferior a 10 segundos da anomalías falsas, si la red a veces se ralentiza e introduce retardos de esta duración.
- Establezca el valor de BATCHHB para reducir la ventana de tiempo para causar un mensaje abandonado debido a que está pendiente en un canal que ha fallado. Es más probable que se produzca un lote pendiente en un canal anómalo si el lote tiene un plazo más largo para llenarse. Si el tráfico de mensajes en el canal es esporádico con largos periodos de tiempo entre ráfagas de mensajes, es más probable que se produzca un lote fallido.
- Surge un problema si el extremo de clúster emisor de un canal falla y, posteriormente, intenta reiniciarse antes de que la pulsación o mantener activo haya detectado el error. El reinicio del canal emisor se rechaza si el extremo de clúster receptor del canal ha permanecido activo. Para evitar la anomalía, disponga que el canal de clúster receptor se termine y se reinicie cuando un canal de clúster emisor intente reiniciarse.

  z/OS **En IBM MQ for z/OS**

  Controle el problema del extremo de clúster receptor del canal que permanece activo utilizando los parámetros **ADOPTMCA** y **ADOPTCHK** en **ALTER QMGR**.

  Multi **En Multiplatforms**

  Controle el problema del extremo de clúster receptor del canal que permanece activo utilizando los atributos **AdoptNewMCA**, **AdoptNewMCATimeout** y **AdoptNewMCACheck** en el archivo qm.ini o el Registro de Windows.

Ejemplo

Consulte “Valores sugeridos” en la página 249 para obtener ejemplos de cómo implementar estos valores en todas las plataformas.

Comprobar que los mandatos asíncronos para redes distribuidos han finalizado

Muchos mandatos son asíncronos cuando se utilizan en una red distribuida. En función del mandato y del estado de la red cuando se emite el mandato, éste puede tardar un periodo de tiempo importante en finalizar. El gestor de colas no emite un mensaje cuando finaliza, por lo tanto, debe comprobar de otro modo si el mandato ha finalizado.

Acerca de esta tarea

Prácticamente cualquier cambio que realice en la configuración de un clúster puede completarse de forma asíncrona. Esto es debido a la administración interna y a los ciclos de actualización que funcionan dentro de los clústeres. En las jerarquías de publicación/suscripción, cualquier cambio de configuración que afecta a las suscripciones tiene la posibilidad de completarse asíncronamente. Esto no siempre resulta obvio a partir del nombre del mandato.

Todos los mandatos MQSC siguientes se pueden completar de forma asíncrona. Cada uno de estos mandatos tiene un equivalente PCF, y la mayor parte también están disponibles desde IBM MQ Explorer. Cuando se ejecutan en una pequeña red sin carga de trabajo, normalmente estos mandatos se completan en pocos segundos. Sin embargo, esto no es así en redes de gran tamaño o muy ocupadas. Además, el mandato **REFRESH CLUSTER** puede tardar mucho más tiempo, en especial cuando se emite en varios gestores de colas al mismo tiempo.

Para estar seguro de que estos mandatos han finalizado, compruebe que existan los objetos previstos en los gestores de colas remotos.

Procedimiento

- ALTER QMGR

  Para el mandato ALTER QMGR PARENT , utilice `DISPLAY PUBSUB TYPE(PARENT) ALL` para realizar un seguimiento del estado de la relación padre solicitada.

  Para los mandatos ALTER QMGR REPOS y ALTER QMGR REPOSNL, utilice `DISPLAY CLUSQMGR QMTYPE` para confirmar la supresión.

- DEFINE CHANNEL, ALTER CHANNEL y DELETE CHANNEL

  Para todos los parámetros que figuran en la tabla Parámetros ALTER CHANNEL, utilice el mandato `DISPLAY CLUSQMGR` para supervisar cuándo se han propagado los cambios en el clúster.

- DEFINE NAMELIST, ALTER NAMELIST y DELETE NAMELIST.

  Si utiliza **NAMELIST** en el atributo **CLUSNL** de un objeto **QMgr**, es posible que una cola o un canal de clúster afecte a dicho objeto. Supervise como corresponda el objeto afectado.

  Los cambios en `SYSTEM.QPUBSUB.QUEUE.NAMELIST` pueden afectar la creación o la cancelación de las suscripciones del proxy en una jerarquía de publicación/suscripción. Utilice el mandato `DISPLAY SUB SUBTYPE(PROXY)` para supervisar esto.

- DEFINE colas, ALTER colas y DELETE colas.

  Para todos los parámetros que figuran en la tabla Parámetros que puede devolver el mandato DISPLAY QUEUE, utilice el mandato `DISPLAY QCLUSTER` para supervisar cuándo se han propagado los cambios en el clúster.

- DEFINE SUB y DELETE SUB

  Cuando define la primera suscripción de una cadena de tema, puede crear suscripciones de proxy en una jerarquía de publicación/suscripción o en un clúster de publicación/suscripción. Del mismo modo,

cuando suprime la última suscripción de una cadena de tema, puede cancelar las suscripciones del proxy en una jerarquía de publicación/suscripción o en un clúster de publicación/suscripción.

Para comprobar que haya finalizado un mandato que define o suprime una suscripción, compruebe si existe la suscripción del proxy prevista en los otros gestores de colas de la red distribuida. Si está utilizando el *direccionamiento directo* en un clúster, compruebe que la suscripción de proxy prevista exista en los otros repositorios parciales del clúster. Si está utilizando el *direccionamiento de host de tema* en un clúster, compruebe que la suscripción de proxy prevista exista en los hosts de temas coincidentes. Utilice el siguiente mandato MQSC:

```
DISPLAY SUB(*) SUBTYPE(PROXY)
```

Utilice la misma comprobación para las llamadas MQI de suscripción y de anulación de suscripción equivalentes, cuando se emitan en un clúster o jerarquía:

- Realice la suscripción utilizando MQSUB.
- Anule la suscripción utilizando MQCLOSE con MQCO_REMOVE_SUB.

- DEFINE TOPIC, ALTER TOPIC y DELETE TOPIC

  Para comprobar si ha finalizado un mandato que defina, alerte o suprima un tema de clúster, visualice el tema en los otros repositorios parciales del clúster (si está utilizando el *direccionamiento directo*) o en los otros hosts de tema (si está utilizando el *direccionamiento de host de tema*).

  Para todos los parámetros que figuran en la tabla Parámetros que puede devolver el mandato DISPLAY TOPIC, utilice el mandato `DISPLAY TCLUSTER` para supervisar cuándo se han propagado los cambios en el clúster.

  Nota:

  - El parámetro **CLUSTER** puede afectar la creación o cancelación de las suscripciones del proxy en un clúster de publicación/suscripción.
  - Los parámetros **PROXYSUB** y **SUBSCOPE** pueden afectar la creación o cancelación de las suscripciones del proxy en una jerarquía de publicación/suscripción o en un clúster de publicación/suscripción.
  - Utilice el mandato `DISPLAY SUB SUBTYPE(PROXYSUB)` para supervisar esto.

- Renovar clúster

  Si está ejecutando el mandato **REFRESH CLUSTER**, sondee la profundidad de la cola de mandatos del clúster. Espere a que llegue a cero y permanezca en cero antes de buscar los objetos.

  1. Utilice el siguiente mandato MQSC para comprobar que la profundidad de la cola de mandatos del clúster sea cero.

     ```
     DISPLAY QL(SYSTEM.CLUSTER.COMMAND.QUEUE) CURDEPTH
     ```

  2. Repita la comprobación hasta que la profundidad de la cola llegue a cero y permanezca en cero en la comprobación siguiente.

  El mandato **REFRESH CLUSTER** elimina y vuelve a crear objetos, y cuando las configuraciones son de gran tamaño, puede tardar mucho tiempo en completarse. Consulte Consideraciones de REFRESH CLUSTER para clústeres de publicación/suscripción.

- RENOVAR QMGR TYPE (PROXYSUB)

  Para comprobar que el mandato **REFRESH QMGR TYPE(PROXYSUB)** ha finalizado, compruebe que las suscripciones del proxy se hayan corregido en otros gestores de colas en la red distribuida. Si está utilizando el *direccionamiento directo* en un clúster, compruebe que las suscripciones de proxy se hayan corregido en los otros repositorios parciales del clúster. Si está utilizando el *direccionamiento de*

host de tema en un clúster, compruebe que las suscripciones de proxy previstas se hayan corregido en los hosts de temas coincidentes. Utilice el siguiente mandato MQSC:

```
DISPLAY SUB(*) SUBTYPE(PROXYSUB)
```

- Restablecer clúster

  Para comprobar que se ha completado el mandato **RESET CLUSTER**, utilice `DISPLAY CLUSQMGR`.
- RESET QMGR TYPE (PUBSUB)

  Para comprobar que se ha completado el mandato **RESET QMGR**, utilice `DISPLAY PUBSUB TYPE(PARENT|CHILD)`.

  Nota: El mandato **RESET QMGR** puede hacer que se cancelen las suscripciones del proxy en una jerarquía de publicación/suscripción o en un clúster de publicación/suscripción. Utilice el mandato `DISPLAY SUB SUBTYPE(PROXYSUB)` para supervisar esto.
- También es posible que desee supervisar las otras colas del sistema que, a medida que los mandatos se van completando y cuando estos se completan, tienden a tener una profundidad de cola de cero.

  Por ejemplo, es posible que desee supervisar la cola `SYSTEM.INTER.QMGR.CONTROL` y la cola `SYSTEM.INTER.QMGR.FANREQ`. Consulte Supervisión del tráfico de suscripciones de proxy en clústeres y Equilibrar productores y consumidores en redes de publicación/suscripción.

Qué hacer a continuación

Si estas comprobaciones no confirman que ha finalizado un mandato asíncrono, se puede producir un error. Para investigarlo, en primer lugar, compruebe el registro del gestor de colas en el que se ha emitido el mandato, a continuación, (para un clúster) compruebe los registros del repositorio completo del clúster.

Referencia relacionada

z/OS Comportamiento asíncrono de los mandatos CLUSTER en z/OS

Direccionamiento de mensajes y desde clústeres

Utilice los alias de colas, los alias de gestor de colas y las definiciones de cola remota para conectar clústeres a gestores de colas externos y otros clústeres.

Para obtener más información sobre el direccionamiento de mensajes a y desde clústeres, consulte los subtemas siguientes:

Conceptos relacionados

Clústeres

Componentes de un clúster

“Alias de gestor de colas y clústeres” en la página 406
Utilice alias de gestor de colas para ocultar el nombre de gestores de colas al enviar mensajes dentro o fuera de un clúster, y para equilibrar la carga de trabajo de los mensajes enviados a un clúster.

“Alias de cola y clústeres” en la página 410
Utilice alias de cola para ocultar el nombre de una cola de clúster, para agrupar en clúster una cola, adoptar atributos diferentes o adoptar controles de acceso diferentes.

“Alias de cola de respuesta y clústeres” en la página 409
Se utiliza una definición de alias de cola de respuesta para especificar nombres alternativos para la información de respuesta. Las definiciones de alias de cola de respuesta se pueden utilizar con clústeres igual que en un entorno de gestión de colas distribuidas.

Tareas relacionadas

“Configuración de un clúster de gestores de colas” en la página 309
Los clústeres proporcionan un mecanismo para interconectar gestores de colas de forma que simplifica la configuración inicial y la gestión continua. Puede definir componentes de clúster, y crear y gestionar los clústeres.

“Configurar un nuevo clúster” en la página 324
Siga estas instrucciones para configurar el clúster de ejemplo. Instrucciones separadas describen la configuración del clúster en TCP/IP, LU 6.2 y con una única cola de transmisión o varias colas de transmisión. Pruebe los trabajos del clúster enviando un mensaje de un gestor de colas a otro.

Referencia relacionada

Comparación de agrupación en clúster y gestión de colas distribuidas

Configurar la solicitud/respuesta a un clúster

Configure una vía de mensajes de solicitud/respuesta desde un gestor de colas fuera de un clúster. Oculte los detalles internos del clúster utilizando un gestor de colas de pasarela como la vía de comunicación hacia y desde el clúster.

Antes de empezar

La Figura 53 en la página 394 muestra un gestor de colas llamado QM3 que está fuera del clúster llamado DEMO. QM3 podría ser un gestor de colas en un producto IBM MQ que no da soporte a clústeres. QM3 aloja una cola llamada Q3, que se define de la manera siguiente:

```
DEFINE QLOCAL(Q3)
```

Dentro del clúster hay dos gestores de colas llamados QM1 y QM2. QM2 aloja una cola de clúster llamada Q2, que se define de la manera siguiente:

```
DEFINE QLOCAL(Q2) CLUSTER(DEMO)
```

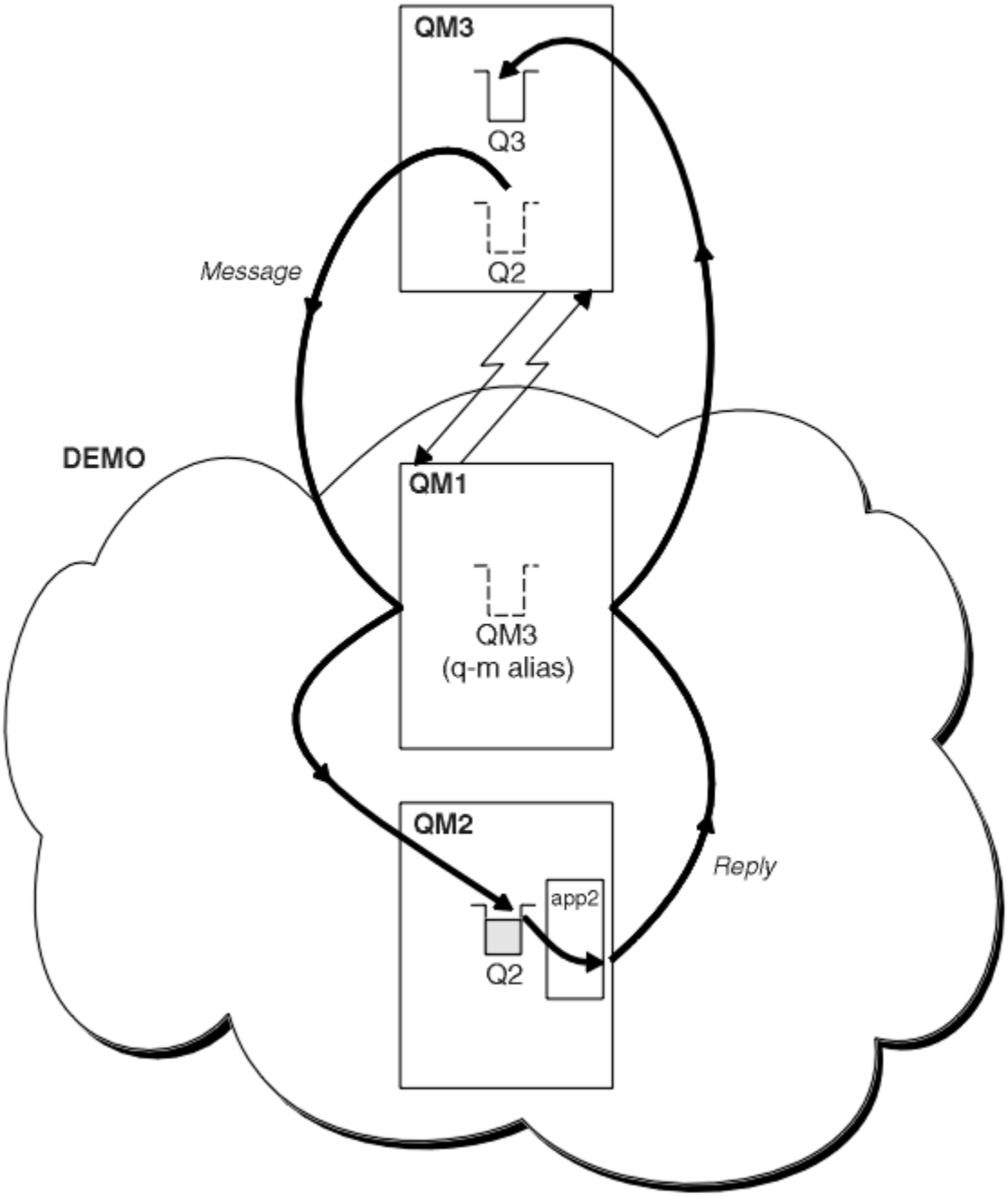


Figura 53. Transferir desde un gestor de colas fuera del clúster

Acerca de esta tarea

Siga los consejos del procedimiento para configurar la vía para los mensajes de solicitud y respuesta.

Procedimiento

1. Envíe el mensaje de solicitud al clúster.

   Tenga en cuenta cómo el gestor de colas que está fuera del clúster transfiere un mensaje a la cola Q2 en QM2, que está dentro del clúster. Un gestor de colas fuera del clúster debe tener una definición QREMOTE para cada cola del clúster a la que transfiere mensajes.

   a) Defina una cola remota para Q2 en QM3.

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(QM2) XMITQ(QM1)
```

Debido a que QM3 no forma parte de un clúster, debe comunicarse utilizando técnicas de gestión de colas distribuidas. Por consiguiente, también debe tener un canal emisor y una cola de transmisión a QM1. QM1 necesita un canal receptor correspondiente. Los canales y las colas de transmisión no se muestran explícitamente en la Figura 53 en la página 394.

En el ejemplo, una aplicación en QM3 emite una llamada MQPUT para transferir un mensaje a Q2. La definición QREMOTE hace que el mensaje se direccione a Q2 en QM2 utilizando el canal emisor que está obteniendo mensajes de la cola de transmisión QM1.

2. Reciba el mensaje de respuesta del clúster.

   Utilice un alias de gestor de colas para crear una vía de retorno para las respuestas a un gestor de colas fuera del clúster. La pasarela QM1, anuncia un alias de gestor de colas para el gestor de colas que está fuera del clúster, QM3. Anuncia QM3 a los gestores de colas dentro del clúster añadiendo el atributo cluster a una definición de alias de gestor de colas para QM3. Una definición de alias de gestor de colas es similar a una definición de cola remota, pero con un RNAME en blanco.

   a) Defina un alias de gestor de colas para QM3 en QM1.

   ```
   DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO)
   ```

   Se ha de considerar detenidamente la elección del nombre para la cola de transmisión utilizada para reenviar las respuestas de nuevo de QM1 a QM3. Implícito en la definición QREMOTE, por la omisión del atributo XMITQ, está el nombre de la cola de transmisión, que es QM3. Pero QM3 es el mismo nombre que tenemos previsto anunciar al resto del clúster utilizando el alias de gestor de colas. IBM MQ no le permite asignar el mismo nombre a la cola de transmisión y al alias de gestor de colas. Una solución es crear una cola de transmisión para reenviar mensajes a QM3 con un nombre diferente del alias de gestor de colas.

   b) Proporcione el nombre de la cola de transmisión en la definición QREMOTE.

   ```
   DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO) XMITQ(QM3.XMIT)
   ```

   El nuevo alias de gestor de colas asocia la nueva cola de transmisión llamada QM3.XMIT con el alias de gestor de colas QM3. Es una solución simple y correcta, pero no totalmente satisfactoria. Se ha infringido el convenio de denominación para colas de transmisión por el que se les asigna el mismo nombre que el gestor de colas de destino. ¿Hay alguna solución alternativa que mantenga el convenio de denominación para colas de transmisión?

   El problema surge porque el solicitante ha pasado de forma predeterminada QM3 como el nombre de gestor de colas de respuesta en el mensaje de solicitud que se envía desde QM3. El servidor en QM2 utiliza el nombre de gestor de colas de respuesta QM3 para dirigir a QM3 en sus respuestas. La solución requería que QM1 anunciara QM3 como el alias de gestor de colas al que devolver los mensajes de respuesta y ha impedido que QM1 utilizara QM3 como el nombre de la cola de transmisión.

   En lugar de proporcionar de forma predeterminada QM3 como el nombre del gestor de colas de respuesta, las aplicaciones en QM3 tienen que pasar un alias de gestor de colas de respuesta a QM1 para los mensajes de respuesta. El gestor de colas de pasarela QM1 anuncia el alias de gestor de colas para respuestas a QM3 en lugar del propio QM3, evitando el conflicto con el nombre de la cola de transmisión.

   c) Defina un alias de gestor de colas para QM3 en QM1.

   ```
   DEFINE QREMOTE(QM3.ALIAS) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO)
   ```

   Es necesario realizar dos cambios en los mandatos de configuración.

i) La QREMOTE en QM1 ahora anuncia el alias de gestor de colas QM3.ALIAS al resto del clúster, asociándolo con el nombre del gestor de colas real QM3. QM3 es de nuevo el nombre de la cola de transmisión para enviar respuestas de nuevo a QM3

ii) La aplicación cliente debe proporcionar QM3.ALIAS como el nombre del gestor de colas de respuesta cuando construye el mensaje de solicitud. Puede proporcionar QM3.ALIAS a la aplicación cliente de una de dos maneras.

- Codifique QM3.ALIAS en el campo de nombre de gestor de colas de respuesta construido por MQPUT en el MQMD. Debe hacerlo de esta manera si está utilizando una cola dinámica para las respuestas.
- Utilice un alias de cola de respuesta, Q3.ALIAS, en lugar de una cola de respuesta al proporcionar el nombre de la cola de respuesta.

```
DEFINE QREMOTE(Q3.ALIAS) RNAME(Q3) RQMNAME(QM3.ALIAS)
```

Qué hacer a continuación

Nota: No puede demostrar el uso de los alias de cola de respuesta con **AMQSREQ0**. Éste abre la cola de respuesta utilizando el nombre de cola proporcionado en el parámetro 3, o la cola modelo SYSTEM.SAMPLE.REPLY predeterminada. Tiene que modificar el ejemplo proporcionando otro parámetro que contenga el alias de cola de respuesta para especificar el alias de gestor de colas de respuesta para MQPUT.

Conceptos relacionados

Alias de gestor de colas y clústeres
Utilice alias de gestor de colas para ocultar el nombre de gestores de colas al enviar mensajes dentro o fuera de un clúster, y para equilibrar la carga de trabajo de los mensajes enviados a un clúster.

Alias de cola de respuesta y clústeres
Se utiliza una definición de alias de cola de respuesta para especificar nombres alternativos para la información de respuesta. Las definiciones de alias de cola de respuesta se pueden utilizar con clústeres igual que en un entorno de gestión de colas distribuidas.

Alias de cola y clústeres
Utilice alias de cola para ocultar el nombre de una cola de clúster, para agrupar en clúster una cola, adoptar atributos diferentes o adoptar controles de acceso diferentes.

Tareas relacionadas

Configurar la solicitud/respuesta desde un clúster
Configure una vía de mensajes de solicitud/respuesta desde un clúster a un gestor de colas fuera del clúster. Oculte los detalles de cómo un gestor de colas dentro del clúster se comunica fuera del clúster utilizando un gestor de colas de pasarela.

Configurar el equilibrio de carga de trabajo desde fuera de un clúster
Configure una vía de mensajes desde un gestor de colas fuera de un clúster a cualquier copia de una cola de clúster. El resultado es equilibrar la carga de trabajo de las solicitudes de fuera del clúster a cada instancia de una cola de clúster.

Configurar vías de acceso de mensajes entre clústeres
Conecte clústeres entre sí utilizando un gestor de colas de pasarela. Haga que las colas o los gestores de colas sean visibles para todos los clústeres definiendo alias de cola de clúster o de gestor de colas de clúster en el gestor de colas de pasarela.

“Ocultar el nombre de un gestor de colas de destino del clúster” en la página 396
Direccione un mensaje a una cola de clúster que esté definida en cualquier gestor de colas en un clúster sin mencionar el gestor de colas.

Ocultar el nombre de un gestor de colas de destino del clúster
Direccione un mensaje a una cola de clúster que esté definida en cualquier gestor de colas en un clúster sin mencionar el gestor de colas.

Antes de empezar

- Evite revelar los nombres de los gestores de colas que están dentro del clúster a los gestores de colas que están fuera del clúster.
  - Resolver las referencias al gestor de colas que aloja una cola dentro del clúster elimina la flexibilidad para realizar el equilibrio de carga de trabajo.
  - También hace que le sea difícil cambiar un gestor de colas que aloja una cola en el clúster.
  - La alternativa es sustituir RQMNAME por un alias de gestor de colas proporcionado por el administrador del clúster.
  - "Ocultar el nombre de un gestor de colas de destino del clúster" en la página 396 describe el uso de un alias de gestor de colas para desasociar un gestor de colas fuera de un clúster de la gestión de gestores de colas dentro de un clúster.
- Sin embargo, la forma recomendada para denominar colas de transmisión es asignarles el nombre del gestor de colas de destino. El nombre de la cola de transmisión revela el nombre de un gestor de colas en el clúster. Debe elegir la regla que desea seguir. Puede elegir denominar la cola de transmisión utilizando el nombre del gestor de colas o el nombre del clúster :

  Denominar la cola de transmisión utilizando el nombre del gestor de colas de pasarela
  La divulgación del nombre del gestor de colas de pasarela a los gestores de colas fuera de un clúster es una excepción razonable a la regla de ocultar nombres de gestor de colas de clúster.

  Denominar la cola de transmisión utilizando el nombre del clúster
  Si no sigue el convenio de denominar las colas de transmisión con el nombre del gestor de colas de destino, utilice el nombre del clúster.

Acerca de esta tarea

Modifique la tarea "Configurar la solicitud/respuesta a un clúster" en la página 393 para ocultar el nombre del gestor de colas de destino dentro del clúster.

Procedimiento

En el ejemplo, consulte la Figura 54 en la página 398, defina un alias de gestor de colas en el gestor de colas de pasarela QM1 llamado DEMO

```
DEFINE QREMOTE(DEMO) RNAME(' ') RQMNAME(' ')
```

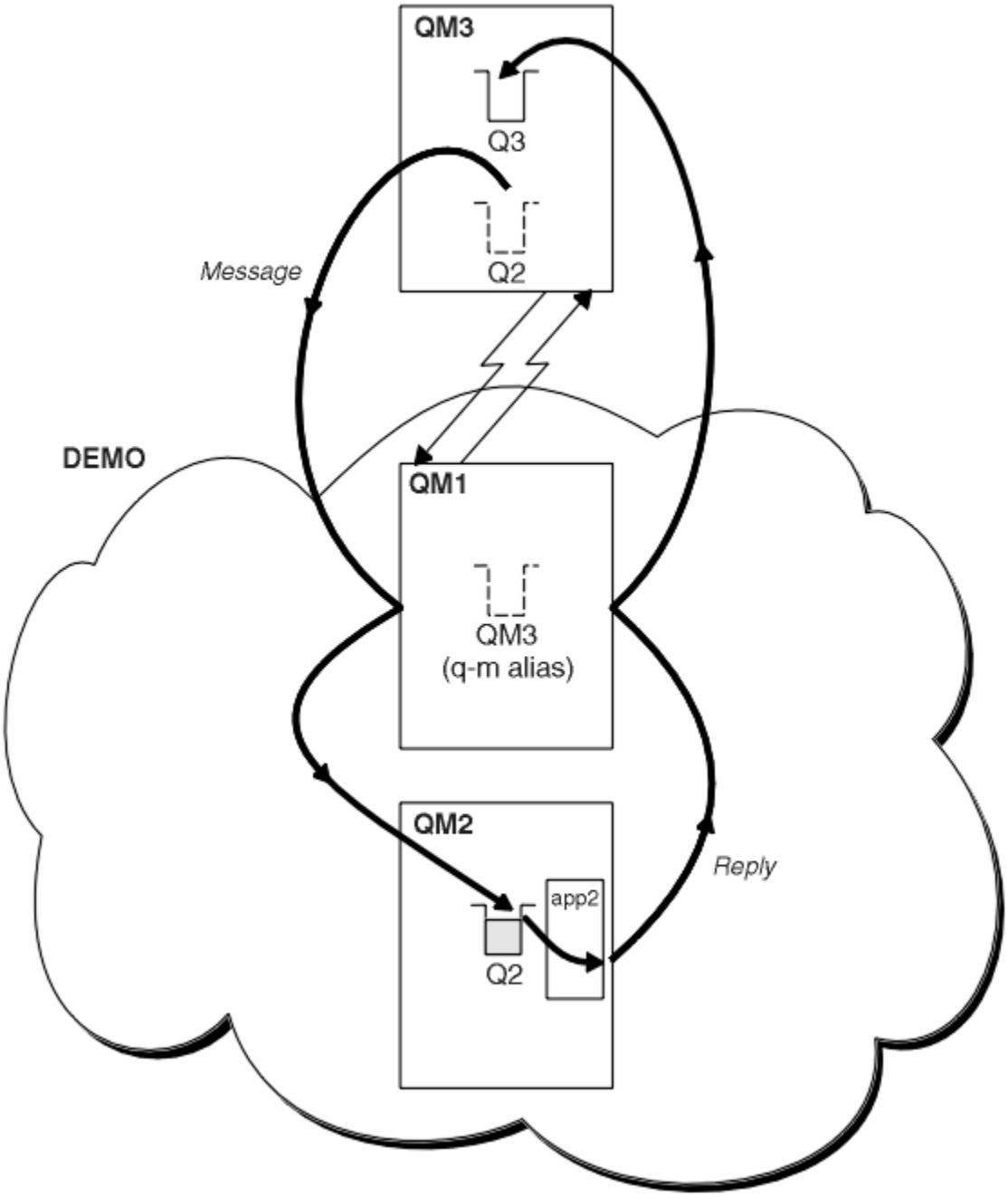


Figura 54. Transferir desde un gestor de colas fuera del clúster

La definición de QREMOTE en QM1 hace que el alias de gestor de colas DEMO sea conocido por el gestor de colas de pasarela. QM3 el gestor de colas fuera del clúster, puede utilizar el alias de gestor de colas DEMO para enviar mensajes a colas de clúster en DEMO, en lugar de tener que utilizar un nombre de gestor de colas real.

Si adopta el convenio de utilizar el nombre de clúster para denominar la cola de transmisión que se conecta a un clúster, entonces la definición de cola remota para Q2 se convierte en:

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(DEMO) XMIT(DEMO)
```

Resultados

Los mensajes destinados para Q2 en DEMO se colocan en la cola de transmisión DEMO. El canal emisor los transfiere de la cola de transmisión al gestor de colas de pasarela, QM1. El gestor de colas de pasarela direcciona los mensajes a cualquier gestor de colas en el clúster que aloja la cola de clúster Q2.

Configurar la solicitud/respuesta desde un clúster

Configure una vía de mensajes de solicitud/respuesta desde un clúster a un gestor de colas fuera del clúster. Oculte los detalles de cómo un gestor de colas dentro del clúster se comunica fuera del clúster utilizando un gestor de colas de pasarela.

Antes de empezar

La Figura 55 en la página 400 muestra un gestor de colas, QM2, dentro del clúster DEMO. Este envía una solicitud a una cola, Q3, alojada en el gestor de colas fuera del clúster. Las respuestas se devuelven a Q2 en QM2 dentro del clúster.

Para comunicar con el gestor de colas fuera del clúster, uno o más gestores de colas dentro del clúster actúan como pasarela. Un gestor de colas de pasarela tiene una vía de comunicación con los gestores de colas fuera del clúster. En el ejemplo, QM1 es la pasarela.

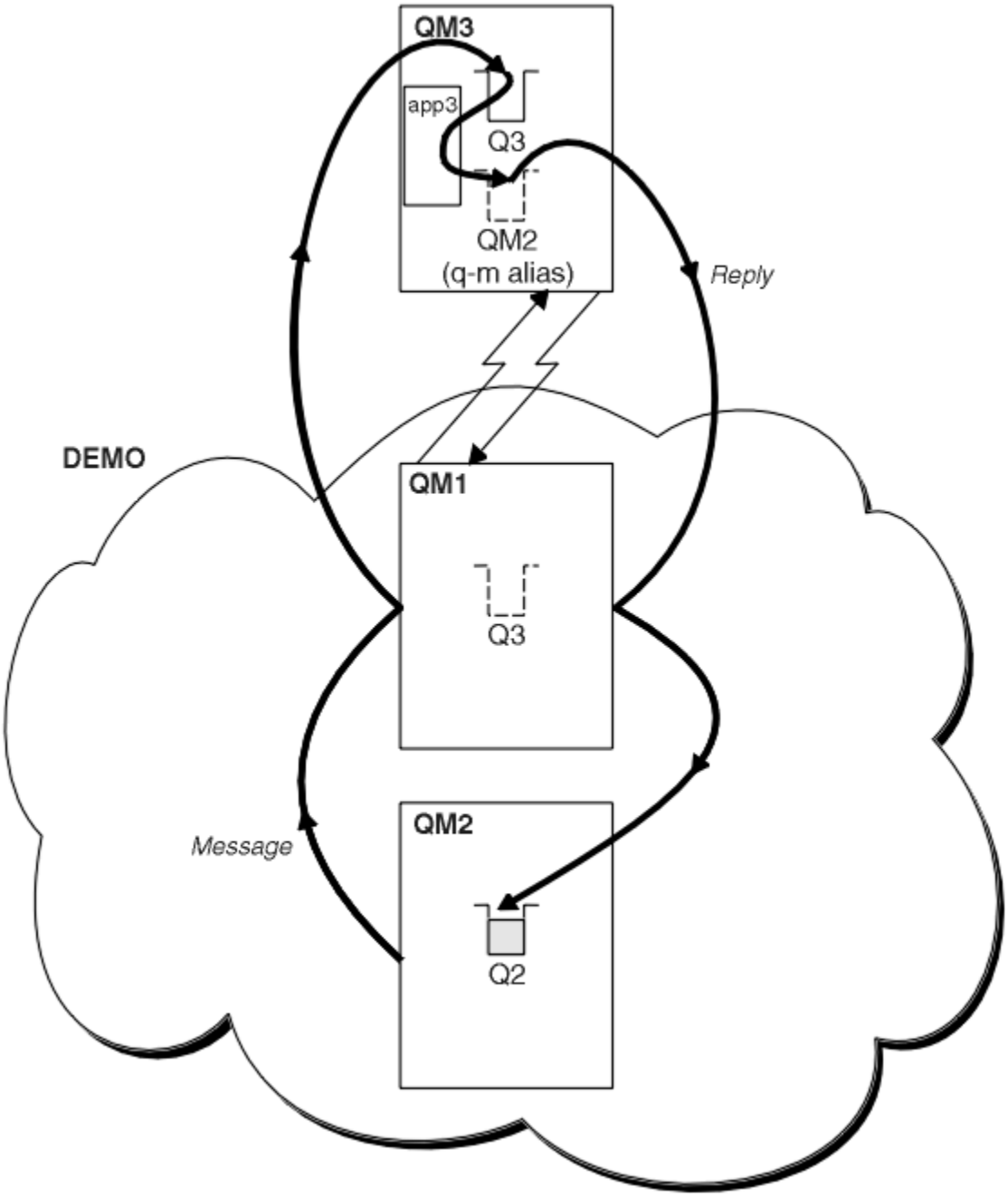


Figura 55. Transferir a un gestor de colas fuera del clúster

Acerca de esta tarea

Siga las instrucciones para configurar la vía para los mensajes de solicitud y respuesta.

Procedimiento

1. Envíe el mensaje de solicitud desde el clúster.

   Tenga en cuenta cómo el gestor de colas, QM2, que está dentro del clúster, transfiere un mensaje a la cola Q3 en QM3, que está fuera del clúster.

   a) Cree una definición QREMOTE en QM1 que anuncie la cola remota Q3 al clúster

```
DEFINE QREMOTE(Q3) RNAME(Q3) RQMNAME(QM3) CLUSTER(DEMO)
```

También tiene un canal emisor y una cola de transmisión al gestor de colas que está fuera del clúster. QM3 tiene un canal receptor correspondiente. Los canales no se muestran en la Figura 55 en la página 400.

Una aplicación en QM2 emite una llamada MQPUT que especifica la cola de destino y la cola a la que se van a enviar las respuestas. La cola de destino es Q3 y la cola de respuesta es Q2.

El mensaje se envía a QM1, que utiliza su definición de cola remota para resolver el nombre de cola en Q3 en QM3.

2. Reciba el mensaje de respuesta del gestor de colas fuera del clúster.

   Un gestor de colas fuera del clúster debe tener un alias de gestor de colas para cada gestor de colas en el clúster al que envía un mensaje. El alias del gestor de colas también debe especificar el nombre de la cola de transmisión al gestor de colas de pasarela. En este ejemplo, QM3 necesita una definición de alias de gestor de colas para QM2:

   a) Cree un alias de gestor de colas QM2 en QM3

   ```
   DEFINE QREMOTE(QM2) RNAME(' ') RQMNAME(QM2) XMITQ(QM1)
   ```

   QM3 también necesita un canal emisor y una cola de transmisión para QM1, y QM1 necesita un canal receptor correspondiente.

   La aplicación, **app3**, en QM3 puede entonces enviar respuestas a QM2, emitiendo una llamada MQPUT y especificando el nombre de cola, Q2, y el nombre de gestor de colas, QM2.

Qué hacer a continuación

Puede definir más de una ruta fuera de un clúster.

Conceptos relacionados

Alias de gestor de colas y clústeres
Utilice alias de gestor de colas para ocultar el nombre de gestores de colas al enviar mensajes dentro o fuera de un clúster, y para equilibrar la carga de trabajo de los mensajes enviados a un clúster.

Alias de cola de respuesta y clústeres
Se utiliza una definición de alias de cola de respuesta para especificar nombres alternativos para la información de respuesta. Las definiciones de alias de cola de respuesta se pueden utilizar con clústeres igual que en un entorno de gestión de colas distribuidas.

Alias de cola y clústeres
Utilice alias de cola para ocultar el nombre de una cola de clúster, para agrupar en clúster una cola, adoptar atributos diferentes o adoptar controles de acceso diferentes.

Tareas relacionadas

Configurar la solicitud/respuesta a un clúster
Configure una vía de mensajes de solicitud/respuesta desde un gestor de colas fuera de un clúster. Oculte los detalles internos del clúster utilizando un gestor de colas de pasarela como la vía de comunicación hacia y desde el clúster.

Configurar el equilibrio de carga de trabajo desde fuera de un clúster
Configure una vía de mensajes desde un gestor de colas fuera de un clúster a cualquier copia de una cola de clúster. El resultado es equilibrar la carga de trabajo de las solicitudes de fuera del clúster a cada instancia de una cola de clúster.

Configurar vías de acceso de mensajes entre clústeres
Conecte clústeres entre sí utilizando un gestor de colas de pasarela. Haga que las colas o los gestores de colas sean visibles para todos los clústeres definiendo alias de cola de clúster o de gestor de colas de clúster en el gestor de colas de pasarela.

Configurar el equilibrio de carga de trabajo desde fuera de un clúster

Configure una vía de mensajes desde un gestor de colas fuera de un clúster a cualquier copia de una cola de clúster. El resultado es equilibrar la carga de trabajo de las solicitudes de fuera del clúster a cada instancia de una cola de clúster.

Antes de empezar

Configure el ejemplo, tal como se muestra en la Figura 53 en la página 394 en "Configurar la solicitud/respuesta a un clúster" en la página 393.

Acerca de esta tarea

En este escenario, el gestor de colas fuera del clúster, QM3 en Figura 56 en la página 403, envía solicitudes a la cola Q2. Q2 está alojada en dos gestores de colas, QM2 y QM4 dentro del clúster DEMO. Los dos gestores de colas están configurados con una opción de enlace predeterminada de NOTFIXED para poder utilizar el equilibrio de carga de trabajo. Las solicitudes de QM3, el gestor de colas fuera del clúster, se envían a cualquiera de las dos instancias de Q2 a través de QM1.

QM3 no forma parte de un clúster y se comunica mediante técnicas de gestión de colas distribuidas. Debe tener un canal emisor y una cola de transmisión a QM1. QM1 necesita un canal receptor correspondiente. Los canales y las colas de transmisión no se muestran explícitamente en la Figura 56 en la página 403.

El procedimiento amplía el ejemplo de la Figura 53 en la página 394 en "Configurar la solicitud/respuesta a un clúster" en la página 393.

Procedimiento

1. Cree una definición QREMOTE para Q2 en QM3.

   ```
   DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(Q3) XMITQ(QM1)
   ```

   Cree una definición QREMOTE para cada cola del clúster a la que QM3 transfiere mensajes.
2. Cree un alias de gestor de colas Q3 en QM1.

   ```
   DEFINE QREMOTE(Q3) RNAME(' ') RQMNAME(' ')
   ```

   Q3 no es un nombre de gestor de colas real. Es el nombre de una definición de alias de gestor de colas en el clúster que equipara el nombre de alias del gestor de colas Q3 con un espacio en blanco, ' '
3. Defina una cola local llamada Q2 en QM2 y QM4.

   ```
   DEFINE QLOCAL(Q2) CLUSTER(DEMO) DEFBIND(NOTFIXED)
   ```
4. QM1, el gestor de colas de pasarela, no tiene definiciones especiales.

Resultados

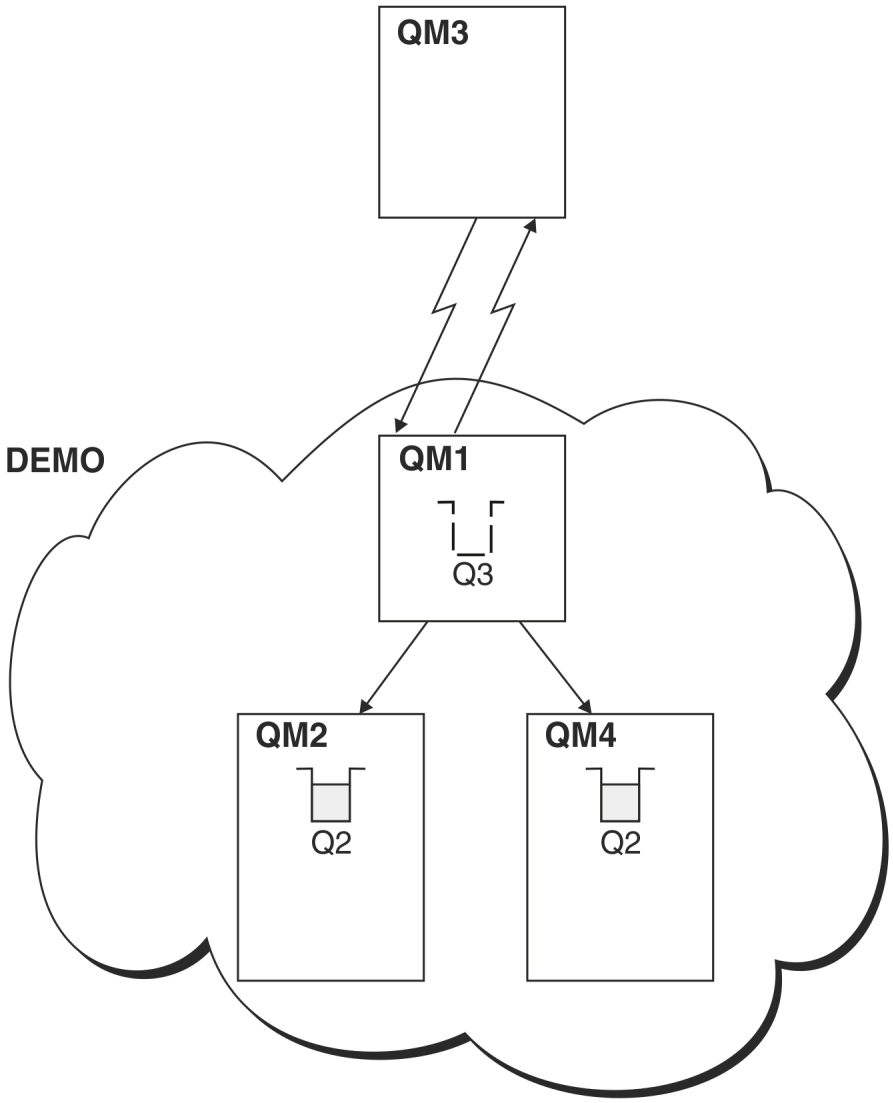


Figura 56. Transferir desde un gestor de colas fuera del clúster

Cuando una aplicación en QM3 emite una llamada MQPUT para transferir un mensaje a Q2, la definición QREMOTE en QM3 hace que el mensaje se direccione a través del gestor de colas de pasarela QM1. Cuando QM1 recibe el mensaje, tiene conocimiento de que el mensaje está destinado a una cola llamada Q2 y realiza la resolución de nombres. QM1 comprueba sus definiciones locales y no encuentra ninguna para Q2. A continuación, QM1 comprueba su configuración de clúster y descubre que tiene conocimiento de dos instancias de Q2 en el clúster DEMO. QM1 ahora puede utilizar el equilibrio de carga de trabajo para distribuir los mensajes entre las instancias de Q2 que residen en QM2 y QM4.

Conceptos relacionados

Alias de gestor de colas y clústeres
Utilice alias de gestor de colas para ocultar el nombre de gestores de colas al enviar mensajes dentro o fuera de un clúster, y para equilibrar la carga de trabajo de los mensajes enviados a un clúster.

Alias de cola de respuesta y clústeres
Se utiliza una definición de alias de cola de respuesta para especificar nombres alternativos para la información de respuesta. Las definiciones de alias de cola de respuesta se pueden utilizar con clústeres igual que en un entorno de gestión de colas distribuidas.

Alias de cola y clústeres

Utilice alias de cola para ocultar el nombre de una cola de clúster, para agrupar en clúster una cola, adoptar atributos diferentes o adoptar controles de acceso diferentes.

Resolución de nombres

Tareas relacionadas

Configurar la solicitud/respuesta a un clúster
Configure una vía de mensajes de solicitud/respuesta desde un gestor de colas fuera de un clúster. Oculte los detalles internos del clúster utilizando un gestor de colas de pasarela como la vía de comunicación hacia y desde el clúster.

Configurar la solicitud/respuesta desde un clúster
Configure una vía de mensajes de solicitud/respuesta desde un clúster a un gestor de colas fuera del clúster. Oculte los detalles de cómo un gestor de colas dentro del clúster se comunica fuera del clúster utilizando un gestor de colas de pasarela.

Configurar vías de acceso de mensajes entre clústeres
Conecte clústeres entre sí utilizando un gestor de colas de pasarela. Haga que las colas o los gestores de colas sean visibles para todos los clústeres definiendo alias de cola de clúster o de gestor de colas de clúster en el gestor de colas de pasarela.

Referencia relacionada

Resolución de nombres de colas

Configurar vías de acceso de mensajes entre clústeres

Conecte clústeres entre sí utilizando un gestor de colas de pasarela. Haga que las colas o los gestores de colas sean visibles para todos los clústeres definiendo alias de cola de clúster o de gestor de colas de clúster en el gestor de colas de pasarela.

Acerca de esta tarea

En lugar de agrupar todos los gestores de colas juntos en un clúster grande, puede tener muchos clústeres más pequeños. Cada clúster tiene uno o más gestores de colas que actúan como puente. La ventaja de esto es que puede restringir la visibilidad de los nombres de cola y de gestor de colas en los clústeres. Consulte Solapamiento de clústeres. Utilice alias para cambiar los nombres de colas y gestores de colas para evitar conflictos de nombres o para cumplir con los convenios de denominación locales.

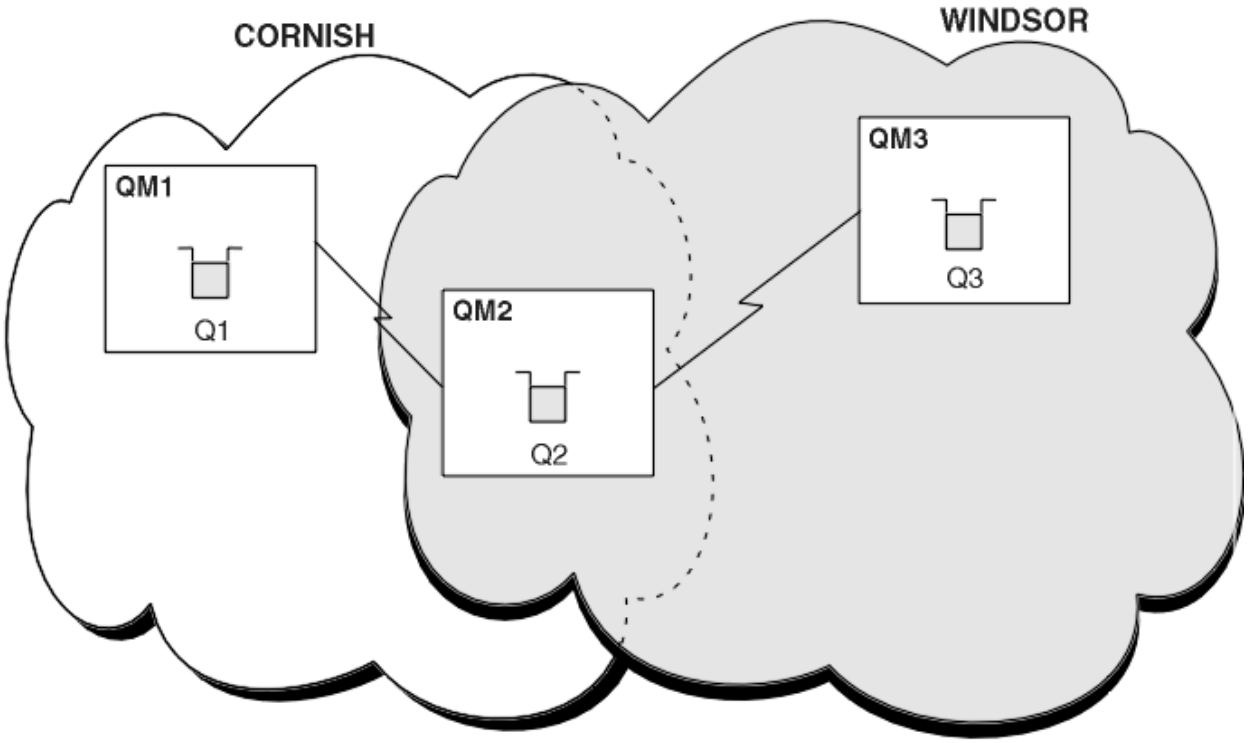


Figura 57. Puente entre clústeres

La Figura 57 en la página 405 muestra dos clústeres con un puente entre ellos. Podría haber más de un puente.

Configure los clústeres realizando el procedimiento siguiente:

Procedimiento

1. Defina una cola de clúster Q1 en QM1.

```
DEFINE QLOCAL(Q1) CLUSTER(CORNISH)
```

2. Defina una cola de clúster Q3 en QM3.

```
DEFINE QLOCAL(Q3) CLUSTER(WINDSOR)
```

3. Cree una lista de nombres llamada CORNISHWINDSOR en QM2 que contenga los nombres de ambos clústeres.

```
DEFINE NAMELIST(CORNISHWINDSOR) DESCR('CornishWindsor namelist')
NAMES(CORNISH, WINDSOR)
```

4. Defina una cola de clúster Q2 en QM2

```
DEFINE QLOCAL(Q2) CLUSNL(CORNISHWINDSOR)
```

Qué hacer a continuación

QM2 es un miembro de ambos clústeres y es el puente entre ellos. Para cada cola que desee hacer visible a través del puente, necesita una definición QALIAS en el puente. Por ejemplo, en la Figura 57 en la página 405, en QM2, necesita:

```
DEFINE QALIAS(MYQ3) TARGET(Q3) CLUSTER(CORNISH) DEFBIND(NOTFIXED)
```

Utilizando el alias de cola, una aplicación conectada a un gestor de colas en CORNISH, por ejemplo QM1, puede transferir un mensaje a Q3. La aplicación hace referencia a Q3 como MYQ3. El mensaje se direcciona a Q3 en QM3.

Cuando se abre una cola, es necesario establecer DEFBIND en NOTFIXED o QDEF. Si DEFBIND se deja en el valor predeterminado, OPEN, el gestor de colas resuelve la definición de alias en el gestor de colas puente que la aloja. El puente no reenvía el mensaje.

Para cada gestor de colas que desea hacer visible, necesita una definición de alias de gestor de colas. Por ejemplo, en QM2, necesita:

```
DEFINE QREMOTE(QM1) RNAME(' ') RQMNAME(QM1) CLUSTER(WINDSOR)
```

Una aplicación conectada a cualquier gestor de colas en WINDSOR, por ejemplo QM3, puede transferir un mensaje a cualquier cola en QM1, nombrando QM1 explícitamente en la llamada MQOPEN.

Conceptos relacionados

Alias de gestor de colas y clústeres
Utilice alias de gestor de colas para ocultar el nombre de gestores de colas al enviar mensajes dentro o fuera de un clúster, y para equilibrar la carga de trabajo de los mensajes enviados a un clúster.

Alias de cola de respuesta y clústeres
Se utiliza una definición de alias de cola de respuesta para especificar nombres alternativos para la información de respuesta. Las definiciones de alias de cola de respuesta se pueden utilizar con clústeres igual que en un entorno de gestión de colas distribuidas.

Alias de cola y clústeres
Utilice alias de cola para ocultar el nombre de una cola de clúster, para agrupar en clúster una cola, adoptar atributos diferentes o adoptar controles de acceso diferentes.

Tareas relacionadas

Configurar la solicitud/respuesta a un clúster
Configure una vía de mensajes de solicitud/respuesta desde un gestor de colas fuera de un clúster. Oculte los detalles internos del clúster utilizando un gestor de colas de pasarela como la vía de comunicación hacia y desde el clúster.

Configurar la solicitud/respuesta desde un clúster
Configure una vía de mensajes de solicitud/respuesta desde un clúster a un gestor de colas fuera del clúster. Oculte los detalles de cómo un gestor de colas dentro del clúster se comunica fuera del clúster utilizando un gestor de colas de pasarela.

Configurar el equilibrio de carga de trabajo desde fuera de un clúster
Configure una vía de mensajes desde un gestor de colas fuera de un clúster a cualquier copia de una cola de clúster. El resultado es equilibrar la carga de trabajo de las solicitudes de fuera del clúster a cada instancia de una cola de clúster.

Alias de gestor de colas y clústeres

Utilice alias de gestor de colas para ocultar el nombre de gestores de colas al enviar mensajes dentro o fuera de un clúster, y para equilibrar la carga de trabajo de los mensajes enviados a un clúster.

Los alias de gestor de colas, que se crean utilizando una definición de cola remota con un RNAME en blanco, tienen cinco usos:

Volver a correlacionar el nombre de gestor de colas al enviar mensajes

Un alias de gestor de colas se puede utilizar para volver a correlacionar el nombre del gestor de colas especificado en una llamada MQOPEN a otro gestor de colas. Este puede ser un gestor de colas de clúster. Por ejemplo, un gestor de colas podría tener la definición de alias de gestor de colas:

```
DEFINE QREMOTE(YORK) RNAME(' ') RQMNAME(CLUSQM)
```

YORK se puede utilizar como un alias para el gestor de colas llamado CLUSQM. Cuando una aplicación en el gestor de colas que ha realizado esta definición coloca un mensaje en el gestor de colas YORK, el gestor de colas local resuelve el nombre en CLUSQM. Si el gestor de colas local no se llama CLUSQM, coloca el mensaje en la cola de transmisión de clúster para trasladarlo a CLUSQM. También cambia la cabecera de transmisión para que indique CLUSQM en lugar de YORK.

Nota: La definición se aplica sólo en el gestor de colas que la realiza. Para anunciar el alias a todo el clúster, debe añadir el atributo CLUSTER a la definición de cola remota. A continuación, los mensajes de otros gestores de colas que estaban destinados a YORK se envían a CLUSQM.

Alterar o especificar la cola de transmisión al enviar mensajes

La asignación de alias se puede utilizar para unir un clúster a un sistema no de clúster. Por ejemplo, los gestores de colas en el clúster ITALY podrían comunicarse con el gestor de colas llamado PALERMO, que está fuera del clúster. Para comunicarse, uno de los gestores de colas del clúster debe actuar como pasarela. Desde el gestor de colas de pasarela, emita el mandato:

```
DEFINE QREMOTE(ROME) RNAME(' ') RQMNAME(PALERMO) XMITQ(X) CLUSTER(ITALY)
```

El mandato es una definición de alias de gestor de colas. Define y anuncia ROME como un gestor de colas a través del cual los mensajes de cualquier gestor de colas del clúster ITALY pueden llegar a su destino en PALERMO mediante múltiples saltos (multi-hop). Los mensajes colocados en una cola abierta con el nombre de gestor de colas establecido en ROME se envían al gestor de colas de pasarela con la definición de alias de gestor de colas. Una vez allí, los mensajes se colocan en la cola de transmisión X y son trasladados por canales no de clúster al gestor de colas PALERMO.

La elección del nombre ROME en este ejemplo no es significativa. Los valores para QREMOTE y RQMNAME pueden ser ambos el mismo.

Determinar el destino al recibir mensajes

Cuando un gestor de colas recibe un mensaje, extrae el nombre de la cola de destino y del gestor de colas de la cabecera de transmisión. Busca una definición de alias de gestor de colas con el mismo nombre que el gestor de colas de la cabecera de transmisión. Si encuentra uno, sustituye RQMNAME de la definición de alias de gestor de colas con el nombre del gestor de colas de la cabecera de transmisión.

Existen dos razones para utilizar un alias de gestor de colas de esta forma:

- Para dirigir mensajes a otro gestor de colas
- Para modificar el nombre de gestor de colas para que sea el mismo que el del gestor de colas local

Utilizar alias de gestor de colas en un gestor de colas de pasarela para direccionar mensajes entre los gestores de colas en distintos clústeres.

Una aplicación puede enviar un mensaje a una cola en un clúster diferente utilizando un alias de gestor de colas. La cola no tiene que ser necesariamente una cola de clúster. La cola se define en un clúster. La aplicación está conectada a un gestor de colas fuera en un clúster diferente. Un gestor de cola de pasarela conecta los dos clústeres. Si la cola no se ha definido como clúster, para que se lleve a cabo el correcto direccionamiento, la aplicación debe abrir la cola utilizando el nombre de cola y un nombre de alias de gestor de colas de clúster. Para obtener un ejemplo de una configuración, consulte “Creación de dos clústeres solapados con un gestor de cola de pasarela” en la página 358, desde donde se obtiene el flujo de mensajes de respuesta que se muestra en la figura 1.

El diagrama muestra el camino que toma el mensaje de respuesta para volver a una cola dinámica temporal, que se llama RQ. La aplicación de servidor, conectada a QM3, abre la cola de respuestas utilizando el nombre del gestor de colas QM2. El nombre del gestor de colas QM2 se define como un alias de gestor de colas en clúster en QM1. QM3 direcciona el mensaje de respuesta a QM1. QM1 direcciona el mensaje a QM2.

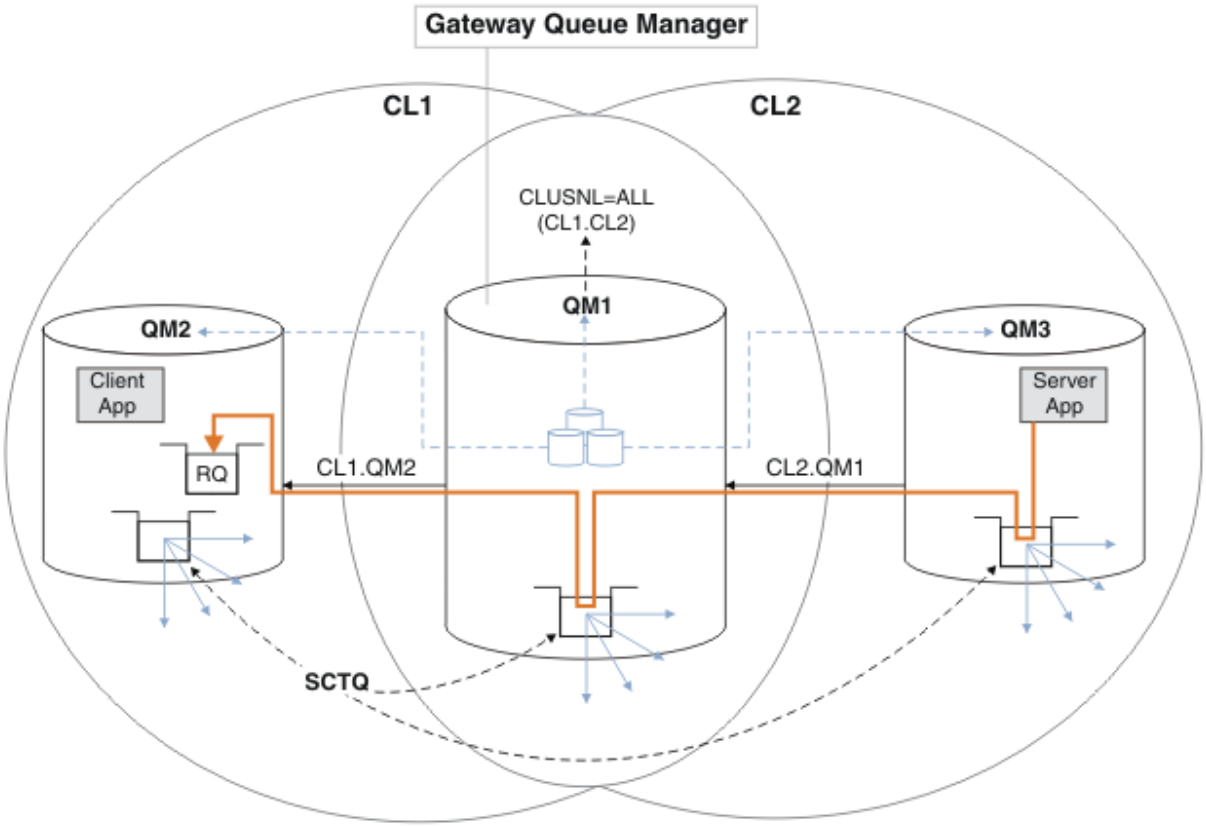


Figura 58. Uso de un alias de gestor de colas para devolver el mensaje de respuesta a un clúster diferente

El modo en que funciona el direccionamiento es el siguiente. Cada gestor de colas de cada clúster tiene una definición de alias de gestor de colas en QM1. Los alias están en clúster en todos los clústeres. Las flechas grises discontinuas de cada uno de los alias a un gestor de colas muestran que cada alias de gestor de colas se resuelve en un gestor de colas real al menos en uno de los clústeres. En este caso, el alias de QM2 se agrupa en el clúster CL1 y CL2, y se resuelve en el gestor de colas real QM2 en CL1. La aplicación de servidor crea el mensaje de respuesta utilizando la respuesta al nombre de cola RQy responde al nombre del gestor de colas QM2. El mensaje se direcciona a QM1 porque la definición de alias del gestor de colas QM2 está definida en QM1 en el clúster CL2 y el gestor de colas QM2 no está en el clúster CL2. Puesto que el mensaje no se puede enviar al gestor de colas de destino, se envía al gestor de colas que tiene la definición de alias.

QM1 coloca el mensaje en la cola de transmisión del clúster en QM1 para transferirlo a QM2. QM1 direcciona el mensaje a QM2 porque la definición de alias del gestor de colas en QM1 para QM2 define QM2 como el gestor de colas de destino real. La definición es no circular, porque las definiciones de alias sólo pueden hacer referencia a definiciones reales; el alias no puede apuntar a sí mismo. QM1 resuelve la definición real, porque tanto QM1 como QM2 están en el mismo clúster, CL1. QM1 averigua la información de conexión de QM2 desde el repositorio para CL1 y direcciona el mensaje a QM2. Para que el mensaje sea redireccionado por QM1, la aplicación de servidor debe haber abierto la cola de respuestas con la opción DEFBIND establecida en MQBND_BIND_NOT_FIXED. Si la aplicación de servidor ha abierto la cola de respuestas con la opción MQBND_BIND_ON_OPEN, el mensaje no se redirecciona y termina en una cola de mensajes no entregados.

Utilizar un gestor de colas como pasarela al clúster para el equilibrio de carga de trabajo para los mensajes procedentes de fuera del clúster.

Define una cola denominada EDINBURGH en más de un gestor de colas en el clúster. Desea que el mecanismo de agrupación en clúster equilibre la carga de trabajo para los mensajes que llegan a esa cola desde fuera del clúster.

Un gestor de colas de fuera del clúster necesita una cola de transmisión y un canal emisor a un gestor de colas del clúster. Esta cola se denomina gestor de colas de pasarela. Para aprovechar las ventajas del mecanismo de equilibrio de carga de trabajo predeterminado, deben aplicarse una de las reglas siguientes:

- El gestor de colas de pasarela no debe contener una instancia de la cola EDINBURGH.
- El gestor de colas de pasarela especifica CLWLUSEQ(ANY) en ALTER QMGR.

Para ver un ejemplo del equilibrio de carga de trabajo desde fuera de un clúster, consulte "Configurar el equilibrio de carga de trabajo desde fuera de un clúster" en la página 402

Conceptos relacionados

Alias de cola de respuesta y clústeres
Se utiliza una definición de alias de cola de respuesta para especificar nombres alternativos para la información de respuesta. Las definiciones de alias de cola de respuesta se pueden utilizar con clústeres igual que en un entorno de gestión de colas distribuidas.

Alias de cola y clústeres
Utilice alias de cola para ocultar el nombre de una cola de clúster, para agrupar en clúster una cola, adoptar atributos diferentes o adoptar controles de acceso diferentes.

Tareas relacionadas

Configurar la solicitud/respuesta a un clúster
Configure una vía de mensajes de solicitud/respuesta desde un gestor de colas fuera de un clúster. Oculte los detalles internos del clúster utilizando un gestor de colas de pasarela como la vía de comunicación hacia y desde el clúster.

Configurar la solicitud/respuesta desde un clúster
Configure una vía de mensajes de solicitud/respuesta desde un clúster a un gestor de colas fuera del clúster. Oculte los detalles de cómo un gestor de colas dentro del clúster se comunica fuera del clúster utilizando un gestor de colas de pasarela.

Configurar el equilibrio de carga de trabajo desde fuera de un clúster
Configure una vía de mensajes desde un gestor de colas fuera de un clúster a cualquier copia de una cola de clúster. El resultado es equilibrar la carga de trabajo de las solicitudes de fuera del clúster a cada instancia de una cola de clúster.

Configurar vías de acceso de mensajes entre clústeres
Conecte clústeres entre sí utilizando un gestor de colas de pasarela. Haga que las colas o los gestores de colas sean visibles para todos los clústeres definiendo alias de cola de clúster o de gestor de colas de clúster en el gestor de colas de pasarela.

Alias de cola de respuesta y clústeres

Se utiliza una definición de alias de cola de respuesta para especificar nombres alternativos para la información de respuesta. Las definiciones de alias de cola de respuesta se pueden utilizar con clústeres igual que en un entorno de gestión de colas distribuidas.

Por ejemplo:

- Una aplicación en el gestor de colas VENICE envía un mensaje al gestor de colas PISA utilizando la llamada MQPUT. La aplicación proporciona la siguiente información de cola de respuesta en el descriptor de mensaje:

```
ReplyToQ='QUEUE'
ReplyToQMgr=''
```

- Para que las respuestas enviadas a QUEUE puedan ser recibidas en OTHERQ en PISA, cree una definición de cola remota en VENICE que se utilice como un alias de cola de respuesta. El alias sólo es efectivo en el sistema en el que se ha creado.

```
DEFINE QREMOTE(QUEUE) RNAME(OTHERQ) RQMNAME(PISA)
```

RQMNAME y QREMOTE pueden especificar el mismo nombre, aunque RQMNAME sea él mismo un gestor de colas de clúster.

Conceptos relacionados

Alias de gestor de colas y clústeres
Utilice alias de gestor de colas para ocultar el nombre de gestores de colas al enviar mensajes dentro o fuera de un clúster, y para equilibrar la carga de trabajo de los mensajes enviados a un clúster.

Alias de cola y clústeres
Utilice alias de cola para ocultar el nombre de una cola de clúster, para agrupar en clúster una cola, adoptar atributos diferentes o adoptar controles de acceso diferentes.

Tareas relacionadas

Configurar la solicitud/respuesta a un clúster
Configure una vía de mensajes de solicitud/respuesta desde un gestor de colas fuera de un clúster. Oculte los detalles internos del clúster utilizando un gestor de colas de pasarela como la vía de comunicación hacia y desde el clúster.

Configurar la solicitud/respuesta desde un clúster
Configure una vía de mensajes de solicitud/respuesta desde un clúster a un gestor de colas fuera del clúster. Oculte los detalles de cómo un gestor de colas dentro del clúster se comunica fuera del clúster utilizando un gestor de colas de pasarela.

Configurar el equilibrio de carga de trabajo desde fuera de un clúster
Configure una vía de mensajes desde un gestor de colas fuera de un clúster a cualquier copia de una cola de clúster. El resultado es equilibrar la carga de trabajo de las solicitudes de fuera del clúster a cada instancia de una cola de clúster.

Configurar vías de acceso de mensajes entre clústeres
Conecte clústeres entre sí utilizando un gestor de colas de pasarela. Haga que las colas o los gestores de colas sean visibles para todos los clústeres definiendo alias de cola de clúster o de gestor de colas de clúster en el gestor de colas de pasarela.

Alias de cola y clústeres

Utilice alias de cola para ocultar el nombre de una cola de clúster, para agrupar en clúster una cola, adoptar atributos diferentes o adoptar controles de acceso diferentes.

Se utiliza una definición QALIAS para crear un alias con el que se conocerá a una cola. Puede crear un alias por una serie de razones:

- Desea empezar a utilizar una cola diferente pero no desea cambiar las aplicaciones.
- No desea que las aplicaciones sepan el nombre real de la cola en la que están colocando mensajes.
- Puede que tenga un convenio de denominación que sea diferente de aquel donde se ha definido la cola.
- Puede que sus aplicaciones no estén autorizadas a acceder a la cola por su nombre real, sino sólo por su alias.

Cree una definición QALIAS en un gestor de colas mediante el mandato DEFINE QALIAS. Por ejemplo, ejecute el mandato:

```
DEFINE QALIAS(PUBLIC) TARGET(LOCAL) CLUSTER(C)
```

El mandato anuncia una cola llamada PUBLIC a los gestores de colas del clúster C. PUBLIC es un alias que se resuelve en la cola denominada LOCAL. Los mensajes enviados a PUBLIC se direccionan a la cola llamada LOCAL.

También puede utilizar una definición de alias de cola para resolver un nombre de cola en una cola de clúster. Por ejemplo, ejecute el mandato:

```
DEFINE QALIAS(PRIVATE) TARGET(PUBLIC)
```

El mandato permite a un gestor de colas utilizar el nombre PRIVATE para acceder a una cola anunciada en otro lugar del clúster con el nombre PUBLIC. Debido a que esta definición no incluye el atributo CLUSTER, sólo se aplica al gestor de colas que la realiza.

Conceptos relacionados

Alias de gestor de colas y clústeres
Utilice alias de gestor de colas para ocultar el nombre de gestores de colas al enviar mensajes dentro o fuera de un clúster, y para equilibrar la carga de trabajo de los mensajes enviados a un clúster.

Alias de cola de respuesta y clústeres
Se utiliza una definición de alias de cola de respuesta para especificar nombres alternativos para la información de respuesta. Las definiciones de alias de cola de respuesta se pueden utilizar con clústeres igual que en un entorno de gestión de colas distribuidas.

Tareas relacionadas

Configurar la solicitud/respuesta a un clúster
Configure una vía de mensajes de solicitud/respuesta desde un gestor de colas fuera de un clúster. Oculte los detalles internos del clúster utilizando un gestor de colas de pasarela como la vía de comunicación hacia y desde el clúster.

Configurar la solicitud/respuesta desde un clúster
Configure una vía de mensajes de solicitud/respuesta desde un clúster a un gestor de colas fuera del clúster. Oculte los detalles de cómo un gestor de colas dentro del clúster se comunica fuera del clúster utilizando un gestor de colas de pasarela.

Configurar el equilibrio de carga de trabajo desde fuera de un clúster
Configure una vía de mensajes desde un gestor de colas fuera de un clúster a cualquier copia de una cola de clúster. El resultado es equilibrar la carga de trabajo de las solicitudes de fuera del clúster a cada instancia de una cola de clúster.

Configurar vías de acceso de mensajes entre clústeres
Conecte clústeres entre sí utilizando un gestor de colas de pasarela. Haga que las colas o los gestores de colas sean visibles para todos los clústeres definiendo alias de cola de clúster o de gestor de colas de clúster en el gestor de colas de pasarela.

Utilización de clústeres para la gestión de carga de trabajo

Al definir varias instancias de una cola en distintos gestores de colas en un clúster, puede extender el trabajo de dar servicio a la cola a través de varios servidores. Existen varios factores que pueden impedir que se vuelvan a poner en cola mensajes en un gestor de colas distintos en el caso de una anomalía.

Así como se pueden configurar clústeres para reducir la administración del sistema, se pueden crear clústeres en los cuales más de un gestor de colas aloje una instancia de la misma cola.

Puede organizar el clúster de forma que los gestores de colas incluidos en el clúster son clones entre sí. Cada gestor de colas puede ejecutar las mismas aplicaciones y tener definiciones locales de las mismas colas. z/OS Por ejemplo, en un sysplex paralelo z/OS, las aplicaciones clonadas pueden acceder a los datos de una base de datos Db2 o VSAM (Método de acceso de almacenamiento virtual) compartida. Puede repartir la carga de trabajo entre los gestores de colas teniendo varias instancias de una aplicación. Cada instancia de la aplicación recibe mensajes y se ejecuta de forma independiente de los otros.

Las ventajas de utilizar los clústeres de esta forma son las siguientes:

- Una mayor disponibilidad de las colas y aplicaciones.
- Un rendimiento más rápido de los mensajes.
- Una mayor distribución de la carga de trabajo en la red.

Cualquiera de los gestores de colas que aloja una instancia de una cola concreta puede manejar mensajes destinados para dicha cola y las aplicaciones no nombran un gestor de colas cuando envían mensajes. Si un clúster contiene más de una instancia de la misma cola, IBM MQ selecciona un gestor de colas al que direccionarle un mensaje. Los destinos apropiados se eligen en función de la disponibilidad del gestor de colas y de la cola, y una serie de atributos específicos de carga de trabajo de clúster asociados a los gestores de colas, las colas y los canales. Consulte Equilibrio de carga de trabajo en clústeres.

z/OS En IBM MQ for z/OS, los gestores de colas que están en grupos de compartición de colas pueden alojar colas de clúster como colas compartidas. Las colas de clúster compartidas están disponibles en todos los gestores de colas del mismo grupo de compartición de colas. Por ejemplo, en la sección Un clúster con varias instancias de la misma cola, uno o ambos de los gestores de colas QM2 y QM4, pueden ser un gestor de colas compartidas. Cada uno tiene una definición para la cola Q3. Cualquiera de los gestores de colas del mismo grupo de compartición de colas como QM4 puede leer los mensajes colocados en la cola compartida Q3. Cada grupo de compartición de colas puede contener hasta 32 gestores de colas, cada uno con acceso a los mismos datos. La compartición de colas aumenta significativamente el rendimiento de los mensajes.

Consulte los subtemas siguientes para obtener más información sobre las configuraciones de clúster para la gestión de cargas:

Conceptos relacionados

Comparación de agrupación en clúster y gestión de colas distribuidas

Gestión de colas distribuidas y clústeres

Componentes de un clúster

Canales de clúster

¿Qué sucede si se inhabilita una cola del clúster para MQPUT?

El equilibrio de la carga de trabajo establecido en el canal de clúster emisor no funciona

"Direccionamiento de mensajes y desde clústeres" en la página 392
Utilice los alias de colas, los alias de gestor de colas y las definiciones de cola remota para conectar clústeres a gestores de colas externos y otros clústeres.

Tareas relacionadas

Escritura y compilación de salidas de carga de trabajo de clúster

"Configuración de un clúster de gestores de colas" en la página 309
Los clústeres proporcionan un mecanismo para interconectar gestores de colas de forma que simplifica la configuración inicial y la gestión continua. Puede definir componentes de clúster, y crear y gestionar los clústeres.

"Configurar un nuevo clúster" en la página 324
Siga estas instrucciones para configurar el clúster de ejemplo. Instrucciones separadas describen la configuración del clúster en TCP/IP, LU 6.2 y con una única cola de transmisión o varias colas de transmisión. Pruebe los trabajos del clúster enviando un mensaje de un gestor de colas a otro.

"Configurar el equilibrio de carga de trabajo desde fuera de un clúster" en la página 402
Configure una vía de mensajes desde un gestor de colas fuera de un clúster a cualquier copia de una cola de clúster. El resultado es equilibrar la carga de trabajo de las solicitudes de fuera del clúster a cada instancia de una cola de clúster.

Referencia relacionada

El programa de ejemplo Cluster Queue Monitoring (AMQSCLM)

Ejemplo de un clúster con más de una instancia de una cola

En este ejemplo de un clúster con más de una instancia de una cola, los mensajes se direccionan a diferentes instancias de la cola. Puede forzar un mensaje en una instancia específica de la cola, y puede elegir enviar una secuencia de mensajes a uno de los dos gestores de colas.

La Figura 59 en la página 413 muestra un clúster en el que hay más de una definición para la cola Q3. Si una aplicación en QM1 transfiere un mensaje a Q3, no necesariamente sabe qué instancia de Q3 va a procesar el mensaje. Si una aplicación se ejecuta en QM2 o QM4, , donde hay instancias locales de Q3, la

instancia local de Q3 se abre de forma predeterminada. Si se establece el atributo de cola CLWLUSEQ, la instancia local de la cola puede tratarse de la misma manera que una instancia remota de la cola.

La opción DefBind de MQOPEN controla si el gestor de colas de destino se selecciona cuando se emite la llamada MQOPEN o cuando el mensaje se transfiere desde la cola de transmisión.

Si establece DefBind en MQBND_BIND_NOT_FIXED, el mensaje se puede enviar a una instancia de la cola que esté disponible cuando se transmita el mensaje. Esto evita los problemas siguientes:

- La cola de destino no está disponible cuando el mensaje llegue al gestor de colas de destino.
- El estado de la cola ha cambiado.
- El mensaje se ha transferido utilizando un alias de cola de clúster, y no existe ninguna instancia de la cola de destino en el gestor de colas donde se ha definido el alias de cola de clúster.

Si se descubre alguno de estos problemas durante la transmisión, se busca otra instancia disponible de la cola de destino y se redirecciona el mensaje. Si ninguna instancia de la cola está disponible, el mensaje se coloca en la cola de mensajes no entregados.

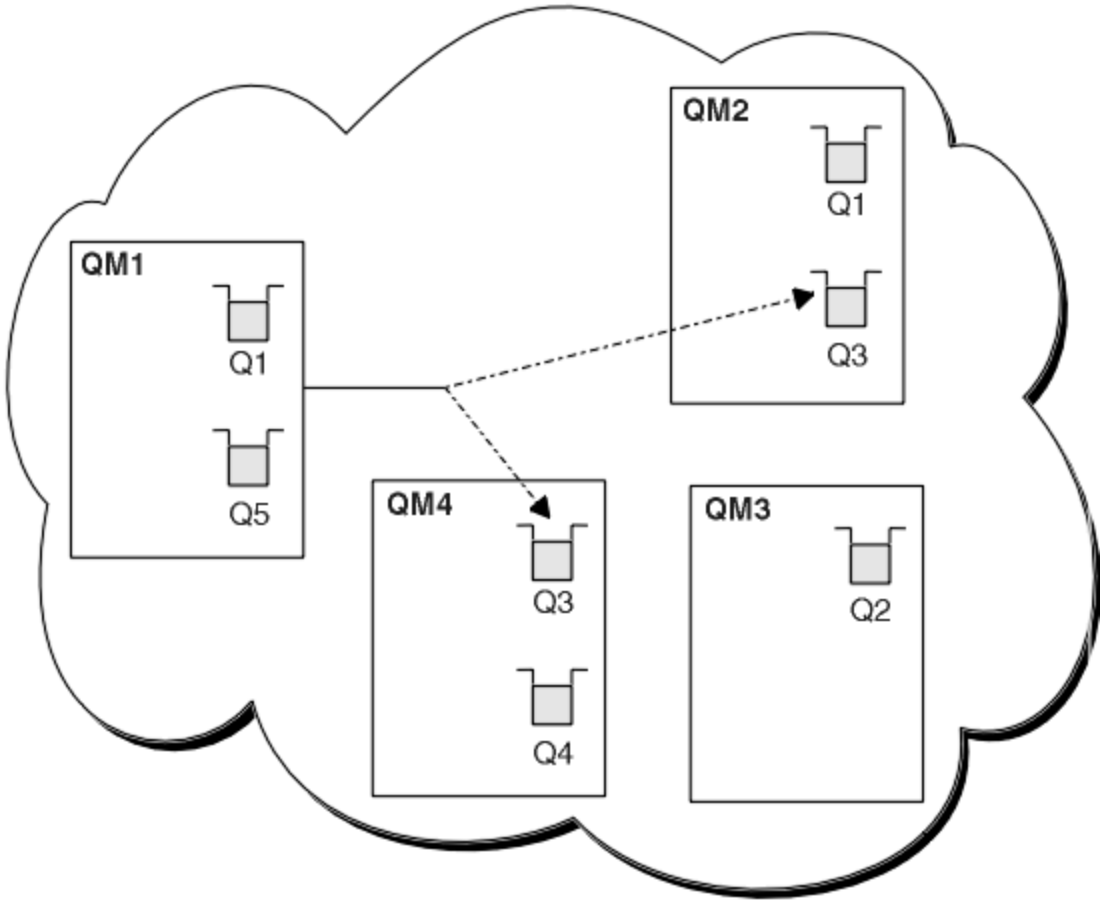


Figura 59. Un clúster con varias instancias de la misma cola

Un factor que puede impedir que los mensajes se redireccionen es si los mensajes se han asignado a un gestor de colas o canal fijo con MQBND_BIND_ON_OPEN. Los mensajes enlazados en MQOPEN no se reasignan nunca a otro canal. Tenga en cuenta que la reasignación de mensajes sólo tiene lugar cuando un canal de clúster está fallando realmente. La reasignación no se produce si el canal ya ha no se realiza correctamente do.

El sistema intenta redireccionar un mensaje si el gestor de colas de destino queda fuera de servicio. Si no lo hace, no afecta a la integridad del mensaje y corre el riesgo de perderlo o crear un duplicado. Si un gestor de colas no se ejecuta correctamente y deja un mensaje pendiente, ese mensaje no se redirecciona.

z/OS En IBM MQ for z/OS, el canal no se detiene completamente hasta que el proceso de reasignación de mensajes se ha completado. Si se detiene el canal con la modalidad establecida en FORCE o TERMINATE, se interrumpirá el proceso; por lo tanto, al hacer esto algunos mensajes BIND_NOT_FIXED se podrían haber ya reasignado a otro canal, o los mensajes podrían estar dañados.

Nota: z/OS

1. Antes de configurar un clúster que tenga varias instancias de la misma cola, asegúrese de que los mensajes no tengan dependencias entre sí. Por ejemplo, que tengan que ser procesados en una secuencia específica o por el mismo gestor de colas.
2. Haga que las definiciones de las diferentes instancias de la misma cola sean idénticas. De lo contrario, obtendrá resultados diferentes de diferentes llamadas MQINQ.

Conceptos relacionados

Programación de aplicaciones y clústeres
No necesita realizar ningún cambio de programación para aprovechar las ventajas de varias instancias de la misma cola. No obstante, algunos programas no funcionan correctamente a menos que se envíe una secuencia de mensajes a la misma instancia de una cola.

Tareas relacionadas

Añadir un gestor de colas que aloja una cola localmente
Siga estas instrucciones para añadir una instancia de INVENTQ para proporcionar capacidad adicional para ejecutar el sistema de aplicación de inventario en París y Nueva York.

Utilizar dos redes en un clúster
Siga estas instrucciones para añadir una nueva tienda en TOKYO, donde hay dos redes diferentes. Ambas deben estar disponibles para comunicarse con el gestor de colas en Tokio.

Utilizar una red primaria y una red secundaria en un clúster
Siga estas instrucciones para hacer que una red sea la red primaria, y otra red sea la red de seguridad. Utilice la red de seguridad si hay un problema con la red primaria.

Añadir una cola para que actúe como copia de seguridad
Siga estas instrucciones para proporcionar una copia de seguridad en Chicago para el sistema de inventario que ahora se ejecuta en Nueva York. El sistema de Chicago sólo se utiliza cuando hay un problema con el sistema de Nueva York.

Restringir el número de canales utilizados
Siga estas instrucciones para restringir el número de canales activos que cada servidor ejecuta cuando se instala una aplicación de consulta de precios en varios gestores de colas.

Añadir un gestor de colas más potente que aloja una cola
Siga estas instrucciones para proporcionar capacidad adicional ejecutando el sistema de inventario en Los Ángeles y en Nueva York, teniendo en cuenta que Los Ángeles puede manejar el doble de mensajes que Nueva York.

Añadir un gestor de colas que aloja una cola localmente

Siga estas instrucciones para añadir una instancia de INVENTQ para proporcionar capacidad adicional para ejecutar el sistema de aplicación de inventario en París y Nueva York.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- El clúster INVENTORY se ha configurado tal como se describe en Añadir un nuevo gestor de colas a un clúster. Contiene tres gestores de colas; LONDON y NEWYORK tienen repositorios completos, PARIS contiene un repositorio parcial. La aplicación de inventario se ejecuta en el sistema de Nueva York,

conectada al gestor de colas NEWYORK. La aplicación se activa con la llegada de mensajes a la cola INVENTQ.

- Queremos añadir una instancia de INVENTQ para proporcionar capacidad adicional para ejecutar el sistema de aplicación de inventario en París y Nueva York.

Acerca de esta tarea

Siga estos pasos para añadir un nuevo gestor de colas que aloje una cola localmente.

Procedimiento

1. Modifique el gestor de colas PARIS.

   Para que la aplicación en París utilice la cola INVENTQ de París y la de Nueva York, debemos informar al gestor de colas. En PARIS, emita el siguiente mandato:

   ```
   ALTER QMGR CLWLUSEQ(ANY)
   ```

2. Revise la aplicación de inventario para ver si tiene afinidades de mensajes.

   Antes de continuar, asegúrese de que la aplicación de inventario no tiene ninguna dependencia de la secuencia de proceso de mensajes. Para obtener más información, consulte Manejo de las afinidades de mensajes.

3. Instale la aplicación de inventario en el sistema en París.
4. Defina la cola de clúster INVENTQ.

   La cola INVENTQ, que ya está alojada por el gestor de colas NEWYORK, también se va a alojar en PARIS. Defínala en el gestor de colas PARIS como se indica a continuación:

   ```
   DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
   ```

   z/OS Ahora que ha completado todas las definiciones, si todavía no lo ha hecho, inicie el iniciador de canal en IBM MQ for z/OS.

   En todas las plataformas, inicie un programa de escucha en el gestor de colas PARIS. El escucha está a la escucha de solicitudes de red entrantes e inicia el canal de clúster receptor cuando es necesario.

Resultados

Figura 60 en la página 416 muestra el clúster configurado por esta tarea.

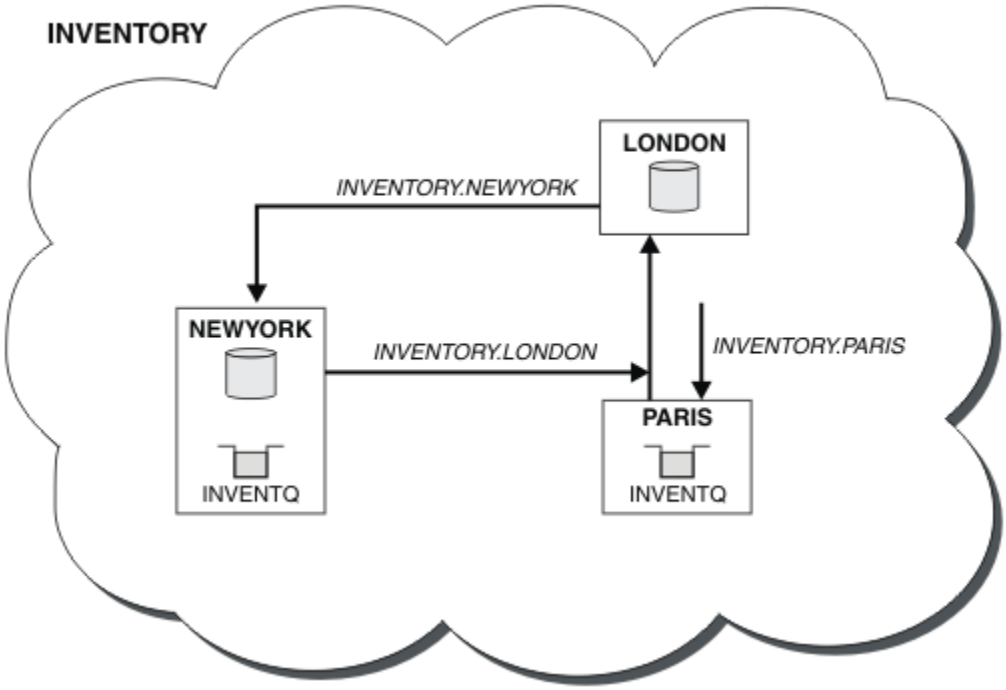


Figura 60. El clúster INVENTORY, con tres gestores de colas

La modificación de este clúster se ha realizado sin tener que alterar los gestores de colas NEWYORK o LONDON. Los repositorios completos en estos gestores de colas se actualizan automáticamente con la información que necesitan para poder enviar mensajes a INVENTQ en PARIS.

Qué hacer a continuación

La cola INVENTQ y la aplicación de inventario ahora están alojadas en dos gestores de colas del clúster. Esto aumenta su disponibilidad, acelera el rendimiento de los mensajes y permite distribuir la carga de trabajo entre los dos gestores de colas. Los mensajes transferidos a INVENTQ por cualquiera de los gestores de colas LONDON, NEWYORK o PARIS se direccionan alternativamente a PARIS o NEWYORK, para equilibrar la carga de trabajo.

Conceptos relacionados

Ejemplo de un clúster con más de una instancia de una cola
En este ejemplo de un clúster con más de una instancia de una cola, los mensajes se direccionan a diferentes instancias de la cola. Puede forzar un mensaje en una instancia específica de la cola, y puede elegir enviar una secuencia de mensajes a uno de los dos gestores de colas.

Programación de aplicaciones y clústeres
No necesita realizar ningún cambio de programación para aprovechar las ventajas de varias instancias de la misma cola. No obstante, algunos programas no funcionan correctamente a menos que se envíe una secuencia de mensajes a la misma instancia de una cola.

Tareas relacionadas

Utilizar dos redes en un clúster
Siga estas instrucciones para añadir una nueva tienda en TOKYO, donde hay dos redes diferentes. Ambas deben estar disponibles para comunicarse con el gestor de colas en Tokio.

Utilizar una red primaria y una red secundaria en un clúster
Siga estas instrucciones para hacer que una red sea la red primaria, y otra red sea la red de seguridad. Utilice la red de seguridad si hay un problema con la red primaria.

Añadir una cola para que actúe como copia de seguridad

Siga estas instrucciones para proporcionar una copia de seguridad en Chicago para el sistema de inventario que ahora se ejecuta en Nueva York. El sistema de Chicago sólo se utiliza cuando hay un problema con el sistema de Nueva York.

Restringir el número de canales utilizados
Siga estas instrucciones para restringir el número de canales activos que cada servidor ejecuta cuando se instala una aplicación de consulta de precios en varios gestores de colas.

Añadir un gestor de colas más potente que aloja una cola
Siga estas instrucciones para proporcionar capacidad adicional ejecutando el sistema de inventario en Los Ángeles y en Nueva York, teniendo en cuenta que Los Ángeles puede manejar el doble de mensajes que Nueva York.

Utilizar dos redes en un clúster

Siga estas instrucciones para añadir una nueva tienda en TOKYO, donde hay dos redes diferentes. Ambas deben estar disponibles para comunicarse con el gestor de colas en Tokio.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- El clúster INVENTORY se ha configurado tal como se describe en "Añadir un nuevo gestor de colas a un clúster". Contiene tres gestores de colas; LONDON y NEWYORK contienen ambos depósitos completos, PARIS contiene un depósito parcial. La aplicación de inventario se ejecuta en el sistema de Nueva York, conectada al gestor de colas NEWYORK. La aplicación se activa con la llegada de mensajes a la cola INVENTQ.
- Se va a añadir una nueva tienda en TOKYO, donde hay dos redes diferentes. Ambas deben estar disponibles para comunicarse con el gestor de colas en Tokio.

Acerca de esta tarea

Siga estos pasos para utilizar dos redes en un clúster.

Procedimiento

1. Decida a qué repositorio completo hace referencia primero TOKYO.

   Cada gestor de colas de un clúster debe hacer referencia a cualquiera de los dos repositorios completos para recopilar información sobre el clúster. De este modo, crea su propio depósito parcial. No tiene mucha importancia qué repositorio elija. En este ejemplo, se elige NEWYORK. Una vez que el nuevo gestor de colas se ha unido al clúster, se comunica con los dos repositorios.
2. Defina los canales CLUSRCVR.

   Cada gestor de colas de un clúster debe definir un clúster receptor en el que pueda recibir mensajes. Este gestor de colas debe poder comunicarse en cada red.

   ```
   DEFINE CHANNEL(INVENTORY.TOKYO.NETB) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME('TOKYO.NETB.CMSTORE.COM') CLUSTER(INVENTORY) DESCR('Cluster-receiver
   channel using network B for TOKYO')

   DEFINE CHANNEL(INVENTORY.TOKYO.NETA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME('TOKYO.NETA.CMSTORE.COM') CLUSTER(INVENTORY) DESCR('Cluster-receiver
   channel using network A for TOKYO')
   ```
3. Defina un canal CLUSSDR en el gestor de colas TOKYO.

Cada gestor de colas de un clúster debe definir un canal de clúster emisor en el que pueda enviar mensajes a su primer repositorio completo. En este caso hemos elegido NEWYORK, por lo que TOKYO necesita la siguiente definición:

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-sender
channel from TOKYO to repository at NEWYORK')
```

z/OS Ahora que ha completado todas las definiciones, si todavía no lo ha hecho, inicie el iniciador de canal en IBM MQ for z/OS.

En todas las plataformas, inicie un programa de escucha en el gestor de colas PARIS. El programa de escucha está a la escucha de peticiones de red entrantes e inicia el canal de clúster receptor cuando es necesario.

Resultados

Figura 61 en la página 418 muestra el clúster configurado por esta tarea.

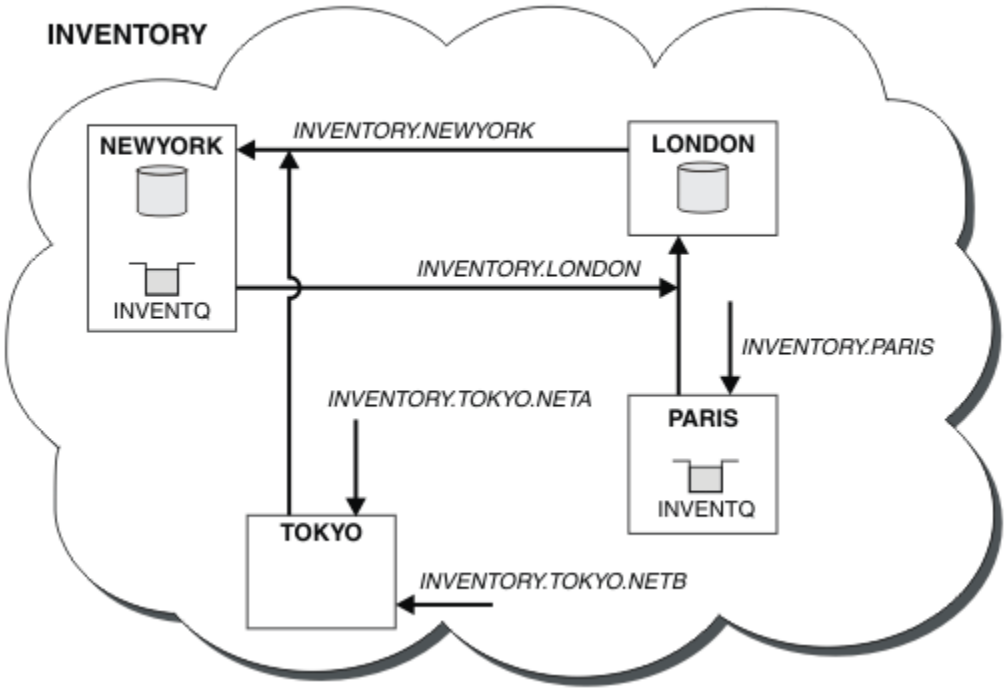


Figura 61. El clúster INVENTORY, con cuatro gestores de colas

Haciendo sólo tres definiciones, hemos añadido el gestor de colas TOKYO al clúster con dos rutas de red diferentes disponibles para el mismo.

Conceptos relacionados

Ejemplo de un clúster con más de una instancia de una cola
En este ejemplo de un clúster con más de una instancia de una cola, los mensajes se direccionan a diferentes instancias de la cola. Puede forzar un mensaje en una instancia específica de la cola, y puede elegir enviar una secuencia de mensajes a uno de los dos gestores de colas.

Programación de aplicaciones y clústeres
No necesita realizar ningún cambio de programación para aprovechar las ventajas de varias instancias de la misma cola. No obstante, algunos programas no funcionan correctamente a menos que se envíe una secuencia de mensajes a la misma instancia de una cola.

Tareas relacionadas

Añadir un gestor de colas que aloja una cola localmente

Siga estas instrucciones para añadir una instancia de INVENTQ para proporcionar capacidad adicional para ejecutar el sistema de aplicación de inventario en París y Nueva York.

Utilizar una red primaria y una red secundaria en un clúster
Siga estas instrucciones para hacer que una red sea la red primaria, y otra red sea la red de seguridad. Utilice la red de seguridad si hay un problema con la red primaria.

Añadir una cola para que actúe como copia de seguridad
Siga estas instrucciones para proporcionar una copia de seguridad en Chicago para el sistema de inventario que ahora se ejecuta en Nueva York. El sistema de Chicago sólo se utiliza cuando hay un problema con el sistema de Nueva York.

Restringir el número de canales utilizados
Siga estas instrucciones para restringir el número de canales activos que cada servidor ejecuta cuando se instala una aplicación de consulta de precios en varios gestores de colas.

Añadir un gestor de colas más potente que aloja una cola
Siga estas instrucciones para proporcionar capacidad adicional ejecutando el sistema de inventario en Los Ángeles y en Nueva York, teniendo en cuenta que Los Ángeles puede manejar el doble de mensajes que Nueva York.

"Añadir un gestor de colas a un clúster" en la página 335
Siga estas instrucciones para añadir un gestor de colas al clúster que ha creado. Los mensajes a temas y colas de clústeres se transfieren utilizando la cola de transmisión de clúster única SYSTEM.CLUSTER.TRANSMIT.QUEUE.

Utilizar una red primaria y una red secundaria en un clúster

Siga estas instrucciones para hacer que una red sea la red primaria, y otra red sea la red de seguridad. Utilice la red de seguridad si hay un problema con la red primaria.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- El clúster de INVENTORY se ha configurado como se describe en "Utilizar dos redes en un clúster" en la página 417. Contiene cuatro gestores de colas; LONDON y NEWYORK contienen ambos depósitos completos; PARIS y TOKYO contienen depósitos parciales. La aplicación de inventario se ejecuta en el sistema de Nueva York, conectada al gestor de colas NEWYORK. El gestor de colas TOKYO tiene dos redes distintas en las que puede comunicarse.
- Quiere hacer que una de las redes sea la red primaria, y que otra de las redes sea la red de seguridad. Tiene previsto utilizar la red de seguridad si hay un problema con la red primaria.

Acerca de esta tarea

Utilice el atributo NETPRTY para configurar una red primaria y una red secundaria en un clúster.

Procedimiento

Modifique los canales CLUSRCVR existentes en TOKYO.

Para indicar que el canal de la red A es el canal primario, y el canal de la red B es el canal secundario, utilice los siguientes mandatos:

a) `ALTER CHANNEL(INVENTORY.TOKYO.NETA) CHLTYPE(CLUSRCVR) NETPRTY(2) DESCR('Main cluster-receiver channel for TOKYO')`

b) `ALTER CHANNEL(INVENTORY.TOKYO.NETB) CHLTYPE(CLUSRCVR) NETPRTY(1) DESCR('Backup cluster-receiver channel for TOKYO')`

Qué hacer a continuación

Al configurar el canal con prioridades de red diferentes, ahora ha definido en el clúster que tiene una red primaria y una red secundaria. Los gestores de colas del clúster que utilicen estos canales utilizarán automáticamente la red primaria siempre que esté disponible. Los gestores de colas harán una sustitución por anomalía para utilizar la red secundaria cuando la red primaria no esté disponible.

Conceptos relacionados

Ejemplo de un clúster con más de una instancia de una cola
En este ejemplo de un clúster con más de una instancia de una cola, los mensajes se direccionan a diferentes instancias de la cola. Puede forzar un mensaje en una instancia específica de la cola, y puede elegir enviar una secuencia de mensajes a uno de los dos gestores de colas.

Programación de aplicaciones y clústeres
No necesita realizar ningún cambio de programación para aprovechar las ventajas de varias instancias de la misma cola. No obstante, algunos programas no funcionan correctamente a menos que se envíe una secuencia de mensajes a la misma instancia de una cola.

Tareas relacionadas

Añadir un gestor de colas que aloja una cola localmente
Siga estas instrucciones para añadir una instancia de INVENTQ para proporcionar capacidad adicional para ejecutar el sistema de aplicación de inventario en París y Nueva York.

Utilizar dos redes en un clúster
Siga estas instrucciones para añadir una nueva tienda en TOKYO, donde hay dos redes diferentes. Ambas deben estar disponibles para comunicarse con el gestor de colas en Tokio.

Añadir una cola para que actúe como copia de seguridad
Siga estas instrucciones para proporcionar una copia de seguridad en Chicago para el sistema de inventario que ahora se ejecuta en Nueva York. El sistema de Chicago sólo se utiliza cuando hay un problema con el sistema de Nueva York.

Restringir el número de canales utilizados
Siga estas instrucciones para restringir el número de canales activos que cada servidor ejecuta cuando se instala una aplicación de consulta de precios en varios gestores de colas.

Añadir un gestor de colas más potente que aloja una cola
Siga estas instrucciones para proporcionar capacidad adicional ejecutando el sistema de inventario en Los Ángeles y en Nueva York, teniendo en cuenta que Los Ángeles puede manejar el doble de mensajes que Nueva York.

Añadir una cola para que actúe como copia de seguridad

Siga estas instrucciones para proporcionar una copia de seguridad en Chicago para el sistema de inventario que ahora se ejecuta en Nueva York. El sistema de Chicago sólo se utiliza cuando hay un problema con el sistema de Nueva York.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- El clúster de INVENTORY se ha configurado como se describe en “Añadir un gestor de colas a un clúster” en la página 335. Contiene tres gestores de colas; LONDON y NEWYORK tienen repositorios completos, PARIS contiene un repositorio parcial. La aplicación de inventario se ejecuta en el sistema de Nueva York, conectada al gestor de colas NEWYORK. La aplicación se activa con la llegada de mensajes a la cola INVENTQ.
- Se está abriendo una nueva tienda en Chicago para proporcionar una copia de seguridad para el sistema de inventario que ahora se ejecuta en Nueva York. El sistema de Chicago sólo se utiliza cuando hay un problema con el sistema de Nueva York.

Acerca de esta tarea

Siga estos pasos para añadir una cola que actúe como copia de seguridad.

Procedimiento

1. Decida a qué repositorio completo hace referencia primero CHICAGO.

   Cada gestor de colas de un clúster debe hacer referencia a cualquiera de los dos repositorios completos para recopilar información sobre el clúster. De este modo, crea su propio depósito parcial. No tiene mucha importancia que repositorio elija para cualquier gestor de colas determinado. En este ejemplo, se elige NEWYORK. Una vez que el nuevo gestor de colas se ha unido al clúster, se comunica con los dos repositorios.
2. Defina el canal CLUSRCVR.

   Cada gestor de colas de un clúster debe definir un clúster receptor en el que pueda recibir mensajes. En CHICAGO, defina:

   ```
   DEFINE CHANNEL(INVENTORY.CHICAGO) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(CHICAGO.CMSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-receiver
   channel for CHICAGO')
   ```
3. Defina un canal CLUSSDR en el gestor de colas CHICAGO.

   Cada gestor de colas de un clúster debe definir un canal de clúster emisor en el que pueda enviar mensajes a su primer repositorio completo. En este caso hemos elegido NEWYORK, por lo que CHICAGO necesita la siguiente definición:

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-sender
   channel from CHICAGO to repository at NEWYORK')
   ```
4. Modifique la cola de clúster existente INVENTQ.

   INVENTQ, que ya está alojada por el gestor de colas NEWYORK, es la instancia principal de la cola.

   ```
   ALTER QLOCAL(INVENTQ) CLWLPRTY(2)
   ```
5. Revise la aplicación de inventario para ver si tiene afinidades de mensajes.

   Antes de continuar, asegúrese de que la aplicación de inventario no tiene ninguna dependencia de la secuencia de proceso de mensajes.
6. Instale la aplicación de inventario en el sistema en CHICAGO.
7. Defina la cola de clúster de seguridad INVENTQ

   INVENTQ que ya está alojada en el gestor de colas NEWYORK, también se va a alojar como copia de seguridad en CHICAGO. Defínala en el gestor de colas CHICAGO como se indica a continuación:

   ```
   DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY) CLWLPRTY(1)
   ```

   z/OS Ahora que ha completado todas las definiciones, si todavía no lo ha hecho, inicie el iniciador de canal en IBM MQ for z/OS.

   En todas las plataformas, inicie un programa de escucha en el gestor de colas CHICAGO. El programa de escucha está a la escucha de peticiones de red entrantes e inicia el canal de clúster receptor cuando es necesario.

Resultados

Figura 62 en la página 422 muestra el clúster configurado por esta tarea.

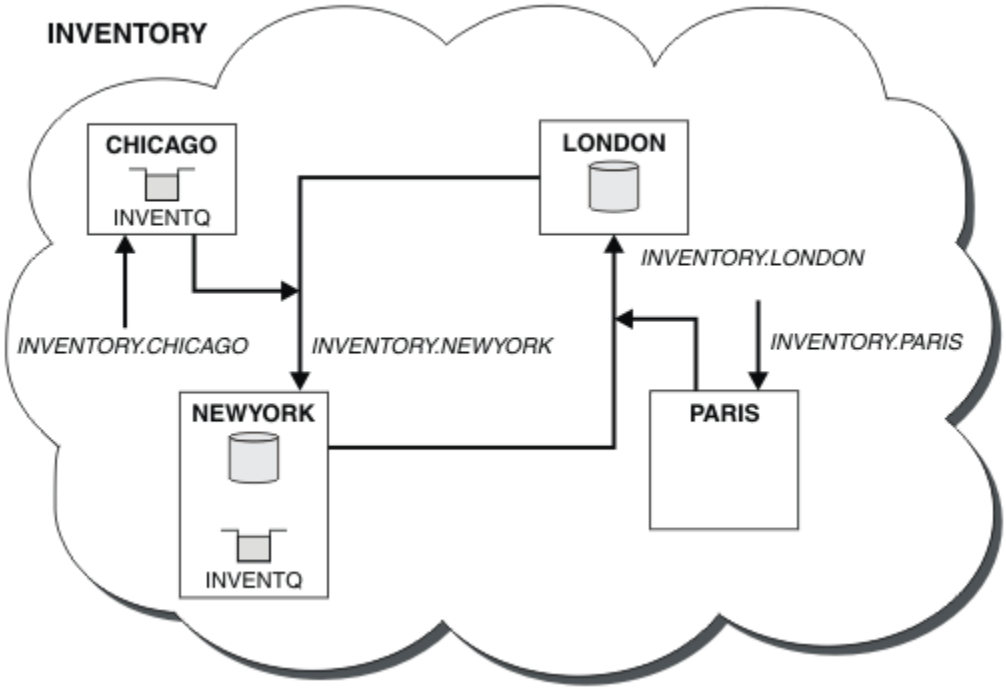


Figura 62. El clúster INVENTORY, con cuatro gestores de colas

La cola INVENTQ y la aplicación de inventario ahora están alojadas en dos gestores de colas del clúster. El gestor de colas CHICAGO es una copia de seguridad. Los mensajes transferidos a INVENTQ se direccionan a NEWYORK, a menos que no esté disponible, en cuyo caso se envían en su lugar a CHICAGO.

Nota:

La disponibilidad de un gestor de colas remoto se basa en el estado del canal que conduce a dicho gestor de colas. Cuando se inician los canales, su estado cambia varias veces; siendo algunos de ellos menos preferibles para el algoritmo de gestión de carga de trabajo de clúster. En la práctica, esto significa que se pueden elegir destinos (de copia de seguridad) de una prioridad menor, mientras se inician los canales que conducen a destinos (principales) de prioridad superior.

Si necesita asegurarse de que ningún mensaje se dirija al destino de copia de seguridad, no utilice CLWLPRTY. Considere la posibilidad de utilizar colas distintas o CLWLRANK con una conmutación manual desde el destino principal al de copia de seguridad.

Conceptos relacionados

Ejemplo de un clúster con más de una instancia de una cola
En este ejemplo de un clúster con más de una instancia de una cola, los mensajes se direccionan a diferentes instancias de la cola. Puede forzar un mensaje en una instancia específica de la cola, y puede elegir enviar una secuencia de mensajes a uno de los dos gestores de colas.

Programación de aplicaciones y clústeres
No necesita realizar ningún cambio de programación para aprovechar las ventajas de varias instancias de la misma cola. No obstante, algunos programas no funcionan correctamente a menos que se envíe una secuencia de mensajes a la misma instancia de una cola.

Tareas relacionadas

Añadir un gestor de colas que aloja una cola localmente
Siga estas instrucciones para añadir una instancia de INVENTQ para proporcionar capacidad adicional para ejecutar el sistema de aplicación de inventario en París y Nueva York.

Utilizar dos redes en un clúster
Siga estas instrucciones para añadir una nueva tienda en TOKYO, donde hay dos redes diferentes. Ambas deben estar disponibles para comunicarse con el gestor de colas en Tokio.

Utilizar una red primaria y una red secundaria en un clúster
Siga estas instrucciones para hacer que una red sea la red primaria, y otra red sea la red de seguridad. Utilice la red de seguridad si hay un problema con la red primaria.

Restringir el número de canales utilizados
Siga estas instrucciones para restringir el número de canales activos que cada servidor ejecuta cuando se instala una aplicación de consulta de precios en varios gestores de colas.

Añadir un gestor de colas más potente que aloja una cola
Siga estas instrucciones para proporcionar capacidad adicional ejecutando el sistema de inventario en Los Ángeles y en Nueva York, teniendo en cuenta que Los Ángeles puede manejar el doble de mensajes que Nueva York.

Restringir el número de canales utilizados

Siga estas instrucciones para restringir el número de canales activos que cada servidor ejecuta cuando se instala una aplicación de consulta de precios en varios gestores de colas.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- Se va a instalar una aplicación de consulta de precios en varios gestores de colas. Para mantener el número de canales utilizados en un número lo más bajo posible, se restringe el número de canales activos que cada servidor ejecuta. La aplicación se activa con la llegada de mensajes a la cola PRICEQ.
- Cuatro gestores de colas de servidor alojan la aplicación de consulta de precios. Dos gestores de colas de consulta envían mensajes a la cola PRICEQ para consultar un precio. Otros dos gestores de colas se configuran como depósitos completos.

Acerca de esta tarea

Siga estos pasos para restringir el número de canales utilizados.

Procedimiento

1. Elija dos depósitos completos.

   Elija dos gestores de colas para que sean los repositorios completos para el clúster de consulta de precios. Estos gestores de colas se llaman REPOS1 y REPOS2.

   Emita el mandato siguiente:

   ```
   ALTER QMGR REPOS(PRICECHECK)
   ```
2. Defina un canal CLUSRCVR en cada gestor de colas.

   En cada gestor de colas del clúster, defina un canal de clúster receptor y un canal de clúster emisor. No importa cuál de ellos se define primero.

   ```
   DEFINE CHANNEL(PRICECHECK.SERVE1) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(SERVER1.COM) CLUSTER(PRICECHECK) DESCR('Cluster-receiver channel')
   ```
3. Defina un canal CLUSSDR en cada gestor de colas.

   Cree una definición CLUSSDR en cada gestor de colas para enlazar ese gestor de colas a cualquiera de los dos gestores de colas de repositorio completo.

   ```
   DEFINE CHANNEL(PRICECHECK.REPOS1) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(REPOS1.COM) CLUSTER(PRICECHECK) DESCR('Cluster-sender channel to
   repository queue manager')
   ```
4. Instale la aplicación de consulta de precios.

5. Defina la cola PRICEQ en todos los gestores de colas de servidor.

   Emita el siguiente mandato en cada uno de ellos:

   ```
   DEFINE QLOCAL(PRICEQ) CLUSTER(PRICECHECK)
   ```

6. Restrinja el número de canales utilizados por las consultas

   En los gestores de colas de consulta, restrinja el número de canales activos utilizados, emitiendo los siguientes mandatos en cada uno de ellos:

   ```
   ALTER QMGR CLWLMRUC(2)
   ```

7. Inicie una sesión de escucha.

   z/OS Si todavía no lo ha hecho, inicie el iniciador de canal en IBM MQ for z/OS.

   En todas las plataformas, inicie un programa de escucha. El programa de escucha está a la escucha de peticiones de red entrantes e inicia el canal de clúster receptor cuando es necesario.

Resultados

Figura 63 en la página 424 muestra el clúster configurado por esa tarea.

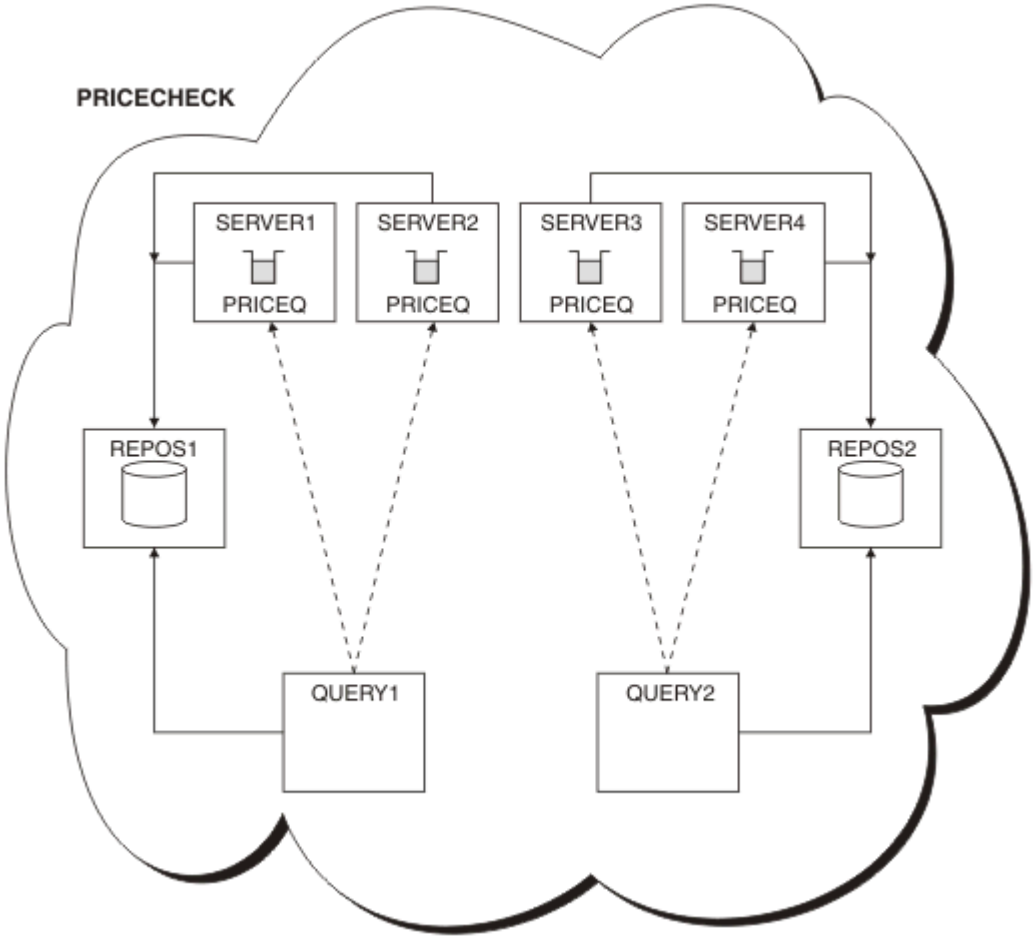


Figura 63. El clúster PRICECHECK, con cuatro gestores de colas de servidor, dos repositorios y dos gestores de colas de consulta

Aunque hay cuatro instancias de la cola PRICEQ disponibles en el clúster PRICECHECK, cada gestor de colas de consulta sólo utiliza dos de ellas. Por ejemplo, el gestor de colas QUERY1 sólo tiene canales activos a los gestores de colas SERVER1 y SERVER2. Si SERVER1 dejara de estar disponible, el gestor de colas QUERY1 empezaría entonces a utilizar otro gestor de colas, por ejemplo SERVER3.

Conceptos relacionados

Ejemplo de un clúster con más de una instancia de una cola
En este ejemplo de un clúster con más de una instancia de una cola, los mensajes se direccionan a diferentes instancias de la cola. Puede forzar un mensaje en una instancia específica de la cola, y puede elegir enviar una secuencia de mensajes a uno de los dos gestores de colas.

Programación de aplicaciones y clústeres
No necesita realizar ningún cambio de programación para aprovechar las ventajas de varias instancias de la misma cola. No obstante, algunos programas no funcionan correctamente a menos que se envíe una secuencia de mensajes a la misma instancia de una cola.

Tareas relacionadas

Añadir un gestor de colas que aloja una cola localmente
Siga estas instrucciones para añadir una instancia de INVENTQ para proporcionar capacidad adicional para ejecutar el sistema de aplicación de inventario en París y Nueva York.

Utilizar dos redes en un clúster
Siga estas instrucciones para añadir una nueva tienda en TOKYO, donde hay dos redes diferentes. Ambas deben estar disponibles para comunicarse con el gestor de colas en Tokio.

Utilizar una red primaria y una red secundaria en un clúster
Siga estas instrucciones para hacer que una red sea la red primaria, y otra red sea la red de seguridad. Utilice la red de seguridad si hay un problema con la red primaria.

Añadir una cola para que actúe como copia de seguridad
Siga estas instrucciones para proporcionar una copia de seguridad en Chicago para el sistema de inventario que ahora se ejecuta en Nueva York. El sistema de Chicago sólo se utiliza cuando hay un problema con el sistema de Nueva York.

Añadir un gestor de colas más potente que aloja una cola
Siga estas instrucciones para proporcionar capacidad adicional ejecutando el sistema de inventario en Los Ángeles y en Nueva York, teniendo en cuenta que Los Ángeles puede manejar el doble de mensajes que Nueva York.

Añadir un gestor de colas más potente que aloja una cola

Siga estas instrucciones para proporcionar capacidad adicional ejecutando el sistema de inventario en Los Ángeles y en Nueva York, teniendo en cuenta que Los Ángeles puede manejar el doble de mensajes que Nueva York.

Antes de empezar

Nota: Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Escenario:

- El clúster de INVENTORY se ha configurado como se describe en “Añadir un gestor de colas a un clúster” en la página 335. Contiene tres gestores de colas: LONDON y NEWYORK contienen ambos depósitos completos, PARIS contiene un depósito parcial y transfiere mensajes desde INVENTQ. La aplicación de inventario se ejecuta en el sistema de Nueva York, conectada al gestor de colas NEWYORK. La aplicación se activa con la llegada de mensajes a la cola INVENTQ.
- Se está abriendo una nueva tienda en Los Ángeles. Para proporcionar capacidad adicional, desea ejecutar el sistema de inventario en Los Ángeles y en Nueva York. El nuevo gestor de colas puede procesar el doble de mensajes que Nueva York.

Acerca de esta tarea

Siga estos pasos para añadir un gestor de colas más potente que aloje una cola.

Procedimiento

1. Decida a qué repositorio completo hace referencia primero LOSANGELES.
2. Cada gestor de colas de un clúster debe hacer referencia a cualquiera de los dos repositorios completos para recopilar información sobre el clúster. De este modo, crea su propio depósito parcial. No tiene mucha importancia qué repositorio elija. En este ejemplo, se elige NEWYORK. Una vez que el nuevo gestor de colas se ha unido al clúster, se comunica con los dos repositorios.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from LOSANGELES to repository at NEWYORK')
   ```

3. Defina el canal CLUSRCVR en el gestor de colas LOSANGELES.

   Cada gestor de colas de un clúster debe definir un canal de clúster receptor en el que puede recibir mensaje. En LOSANGELES, defina:

   ```
   DEFINE CHANNEL(INVENTORY.LOSANGELES) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(LOSANGELES.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-receiver channel for queue manager LOSANGELES')
   CLWLWGHT(2)
   ```

   El canal de clúster receptor anuncia la disponibilidad del gestor de colas para recibir mensajes de otros gestores de colas en el clúster INVENTORY. Si establece CLWLWGHT en dos, se segura de que el gestor de Los Ángeles recibe el doble de mensajes de inventario que Nueva York (cuando el canal para NEWYORK se ha establecido en uno).
4. Modifique el canal CLUSRCVR en el gestor de colas NEWYORK.

   Asegúrese de que el gestor de colas Los Angeles reciba el doble de mensajes de inventario que New York. Modifique la definición del canal de clúster receptor.

   ```
   ALTER CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) CLWLWGHT(1)
   ```

5. Revise la aplicación de inventario para ver si tiene afinidades de mensajes.

   Antes de continuar, asegúrese de que la aplicación de inventario no tiene ninguna dependencia de la secuencia de proceso de mensajes.
6. Instale la aplicación de inventario en el sistema de Los Ángeles.
7. Defina la cola de clúster INVENTQ.

   La cola INVENTQ, que ya está alojada en el gestor de colas NEWYORK, también se va a alojar en LOSANGELES. Defínala en el gestor de colas LOSANGELES como se indica a continuación:

   ```
   DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
   ```

   z/OS Ahora que ha completado todas las definiciones, si todavía no lo ha hecho, inicie el iniciador de canal en IBM MQ for z/OS.

   En todas las plataformas, inicie un programa de escucha en el gestor de colas LOSANGELES. El programa de escucha está a la escucha de peticiones de red entrantes e inicia el canal de clúster receptor cuando es necesario.

Resultados

“Añadir un gestor de colas más potente que aloja una cola” en la página 425 muestra el clúster configurado por esa tarea.

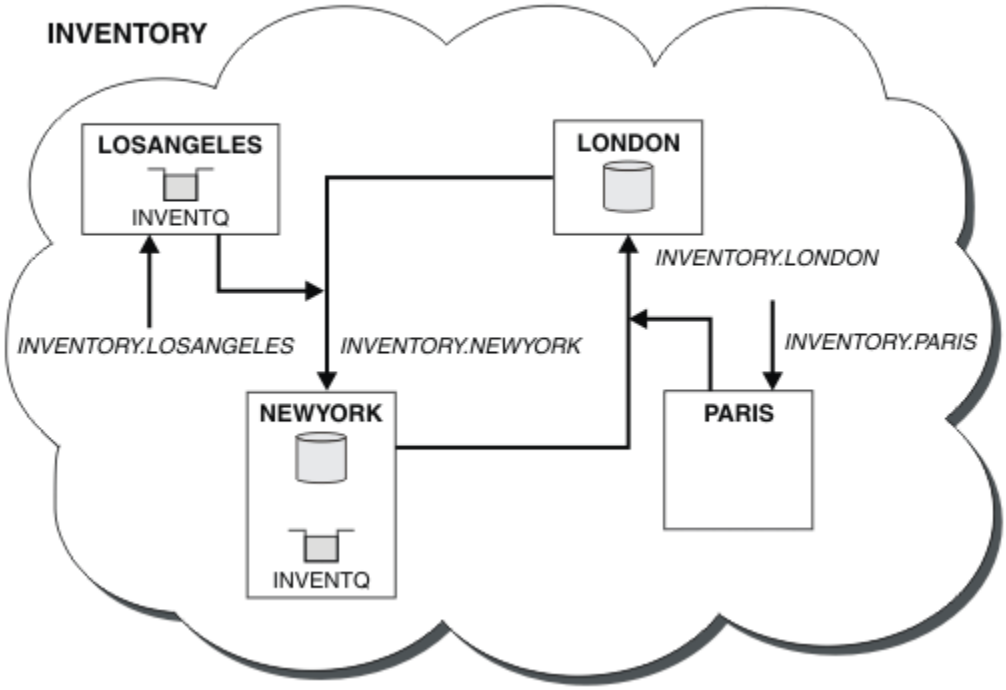


Figura 64. El clúster INVENTORY con cuatro gestores de colas

Esta modificación en el clúster se ha llevado a cabo sin tener que alterar los gestores de colas LONDON y PARIS. Los repositorios en estos gestores de colas se actualizan automáticamente con la información que necesitan para poder enviar mensajes a INVENTQ en LOSANGELES.

Qué hacer a continuación

La cola INVENTQ y la aplicación de inventario están alojadas en dos gestores de colas del clúster. La configuración aumenta su disponibilidad, acelera el rendimiento de los mensajes y permite distribuir la carga de trabajo entre los dos gestores de colas. Los mensajes transferidos a INVENTQ por LOSANGELES o NEWYORK son manejados por la instancia en el gestor de colas local siempre que sea posible. Los mensajes transferidos por LONDON o PARIS se direccionan a LOSANGELES o NEWYORK, enviando el doble de mensajes a LOSANGELES.

Conceptos relacionados

Ejemplo de un clúster con más de una instancia de una cola
En este ejemplo de un clúster con más de una instancia de una cola, los mensajes se direccionan a diferentes instancias de la cola. Puede forzar un mensaje en una instancia específica de la cola, y puede elegir enviar una secuencia de mensajes a uno de los dos gestores de colas.

Programación de aplicaciones y clústeres
No necesita realizar ningún cambio de programación para aprovechar las ventajas de varias instancias de la misma cola. No obstante, algunos programas no funcionan correctamente a menos que se envíe una secuencia de mensajes a la misma instancia de una cola.

Tareas relacionadas

Añadir un gestor de colas que aloja una cola localmente
Siga estas instrucciones para añadir una instancia de INVENTQ para proporcionar capacidad adicional para ejecutar el sistema de aplicación de inventario en París y Nueva York.

Utilizar dos redes en un clúster
Siga estas instrucciones para añadir una nueva tienda en TOKYO, donde hay dos redes diferentes. Ambas deben estar disponibles para comunicarse con el gestor de colas en Tokio.

Utilizar una red primaria y una red secundaria en un clúster

Siga estas instrucciones para hacer que una red sea la red primaria, y otra red sea la red de seguridad. Utilice la red de seguridad si hay un problema con la red primaria.

Añadir una cola para que actúe como copia de seguridad
Siga estas instrucciones para proporcionar una copia de seguridad en Chicago para el sistema de inventario que ahora se ejecuta en Nueva York. El sistema de Chicago sólo se utiliza cuando hay un problema con el sistema de Nueva York.

Restringir el número de canales utilizados
Siga estas instrucciones para restringir el número de canales activos que cada servidor ejecuta cuando se instala una aplicación de consulta de precios en varios gestores de colas.

Programación de aplicaciones y clústeres

No necesita realizar ningún cambio de programación para aprovechar las ventajas de varias instancias de la misma cola. No obstante, algunos programas no funcionan correctamente a menos que se envíe una secuencia de mensajes a la misma instancia de una cola.

Las aplicaciones pueden abrir una cola utilizando la llamada `MQOPEN`. Las aplicaciones utilizan la llamada `MQPUT` para transferir mensajes a una cola abierta. Las aplicaciones pueden transferir un mensaje individual a una cola que aún no está abierta, utilizando la llamada `MQPUT1`.

Si configura clústeres que tienen varias instancias de la misma cola, no hay consideraciones de programación de aplicaciones específicas. Sin embargo, para beneficiarse de los aspectos de la gestión de carga de trabajo de la agrupación en clúster, es posible que tenga que modificar las aplicaciones. Si configura una red en la que hay varias definiciones de la misma cola, revise las aplicaciones para ver si tienen afinidades de mensajes.

Supongamos, por ejemplo, que tiene dos aplicaciones que se basan en una serie de mensajes que fluyen entre ellos en forma de preguntas y respuestas. Es probable que desee que las respuestas se devuelvan al mismo gestor de colas que envió una pregunta. Es importante que la rutina de gestión de carga de trabajo no envíe los mensajes a ningún gestor de colas que aloje una copia de la cola de respuesta.

Es posible que tenga aplicaciones que requieran que los mensajes se procesen secuencialmente (por ejemplo, una aplicación de duplicación de base de datos que envía lotes de mensajes que deben recuperarse secuencialmente). El uso de mensajes segmentados también puede causar un problema de afinidad.

Abrir una versión local o remota de la cola de destino

Tenga en cuenta el modo en que el gestor de colas elige utilizar una versión local o remota de la cola de destino.

1. El gestor de colas abre la versión local de la cola de destino para leer mensajes, o para establecer los atributos de la cola.
2. El gestor de colas abre cualquier instancia de la cola de destino para grabar mensajes, si se da al menos una de las siguientes condiciones:
   - No existe una versión local de la cola de destino.
   - El gestor de colas especifica `CLWLUSEQ(ANY)` en `ALTER QMGR`.
   - La cola en el gestor de colas especifica `CLWLUSEQ(ANY)`.

Conceptos relacionados

Ejemplo de un clúster con más de una instancia de una cola
En este ejemplo de un clúster con más de una instancia de una cola, los mensajes se direccionan a diferentes instancias de la cola. Puede forzar un mensaje en una instancia específica de la cola, y puede elegir enviar una secuencia de mensajes a uno de los dos gestores de colas.

Tareas relacionadas

Añadir un gestor de colas que aloja una cola localmente
Siga estas instrucciones para añadir una instancia de `INVENTQ` para proporcionar capacidad adicional para ejecutar el sistema de aplicación de inventario en París y Nueva York.

Utilizar dos redes en un clúster
Siga estas instrucciones para añadir una nueva tienda en TOKYO, donde hay dos redes diferentes. Ambas deben estar disponibles para comunicarse con el gestor de colas en Tokio.

Utilizar una red primaria y una red secundaria en un clúster
Siga estas instrucciones para hacer que una red sea la red primaria, y otra red sea la red de seguridad. Utilice la red de seguridad si hay un problema con la red primaria.

Añadir una cola para que actúe como copia de seguridad
Siga estas instrucciones para proporcionar una copia de seguridad en Chicago para el sistema de inventario que ahora se ejecuta en Nueva York. El sistema de Chicago sólo se utiliza cuando hay un problema con el sistema de Nueva York.

Restringir el número de canales utilizados
Siga estas instrucciones para restringir el número de canales activos que cada servidor ejecuta cuando se instala una aplicación de consulta de precios en varios gestores de colas.

Añadir un gestor de colas más potente que aloja una cola
Siga estas instrucciones para proporcionar capacidad adicional ejecutando el sistema de inventario en Los Ángeles y en Nueva York, teniendo en cuenta que Los Ángeles puede manejar el doble de mensajes que Nueva York.

Manejo de las afinidades de mensajes
Las afinidades de mensajes rara vez son parte de un buen diseño de programación. Necesita eliminar totalmente las afinidades de mensajes para utilizar clústeres. Si no puede eliminar las afinidades de mensajes, puede forzar a que los mensajes relacionados sean entregados utilizando el mismo canal y al mismo gestor de colas.

Si tiene aplicaciones con afinidades de mensajes, elimine las afinidades antes de empezar a utilizar clústeres.

La eliminación de las afinidades de mensajes mejora la disponibilidad de las aplicaciones. Una aplicación envía un lote de mensajes que tiene afinidades de mensajes a un gestor de colas. El gestor de colas falla después de recibir sólo parte del lote. El gestor de colas emisor debe esperar a que éste se recupere y procese el lote de mensajes incompleto antes de poder enviar más mensajes.

La eliminación de las afinidades de mensajes también mejora la escalabilidad de las aplicaciones. Un lote de mensajes con afinidades puede bloquear recursos en el gestor de colas de destino mientras espera mensajes subsiguientes. Estos recursos pueden permanecer bloqueados durante largos períodos de tiempo, impidiendo que otras aplicaciones realicen su trabajo.

Además, las afinidades de mensajes impiden que las rutinas de gestión de carga de trabajo del clúster hagan la mejor elección del gestor de colas.

Para eliminar afinidades, tenga en cuenta las siguientes posibilidades:

- Transportar información de estado en los mensajes
- Mantener la información de estado en almacenamiento no volátil al que pueda acceder cualquier gestor de colas, por ejemplo en una base de datos Db2
- Replicar datos de sólo lectura para que sean accesibles para más un gestor de colas

Si no es conveniente modificar las aplicaciones para eliminar afinidades de mensajes, hay una serie de posibles soluciones al problema.

Especificar un destino específico en la llamada MQOPEN

Si especifica el nombre de cola remota y el nombre de gestor de colas en cada llamada MQOPEN, todos los mensajes que transfieren a la cola utilizando ese manejador de objeto van al mismo gestor de colas, que puede ser el gestor de colas local.

Especificar el nombre de cola remota y el nombre de gestor de colas en cada llamada MQOPEN tiene desventajas:

- No se lleva a cabo equilibrio de carga de trabajo. No puede aprovechar las ventajas del equilibrio de carga de trabajo del clúster.
- Si el gestor de colas de destino es remoto y hay más de un canal al mismo, los mensajes pueden seguir rutas diferentes y la secuencia de mensajes no se conserva.
- Si su gestor de colas tiene una definición para una cola de transmisión con el mismo nombre que el gestor de colas de destino, los mensajes se colocan en esa cola de transmisión en lugar de en la cola de transmisión de clúster.

Devuelve el nombre del gestor de colas en el campo del gestor de colas de respuesta

Permita que el gestor de colas que recibe el primer mensaje de un lote devuelva su nombre en la respuesta. Para ello, utiliza el campo `ReplyToQMgr` del descriptor de mensaje. El gestor de colas en el extremo emisor puede entonces extraer el nombre del gestor de colas de respuesta y especificarlo en todos los mensajes subsiguientes.

Utilizar la información de `ReplyToQMgr` de la respuesta tiene desventajas:

- El gestor de colas solicitante debe esperar una respuesta a su primer mensaje
- Debe escribir código adicional para buscar y utilizar la información de `ReplyToQMgr` antes de enviar mensajes posteriores
- Si hay más de una ruta al gestor de colas, puede que la secuencia de los mensajes no se conserve

Establecer la opción MQOO_BIND_ON_OPEN en la llamada MQOPEN

Fuerce a que todos los mensajes se coloquen en el mismo destino utilizando la opción `MQOO_BIND_ON_OPEN` en la llamada `MQOPEN`. Se debe especificar `MQOO_BIND_ON_OPEN` o `MQOO_BIND_ON_GROUP` cuando se utilizan grupos de mensajes con clústeres para asegurarse de que todos los mensajes del grupo se procesan en el mismo destino.

Al abrir una cola y especificar `MQOO_BIND_ON_OPEN`, fuerza a que todos los mensajes que se envían a esta cola se envíen a la misma instancia de la cola. `MQOO_BIND_ON_OPEN` enlaza todos los mensajes con el mismo gestor de colas y también con la misma ruta. Por ejemplo, si hay una ruta IP y una ruta NetBIOS al mismo destino, se selecciona una de ellas cuando se abre la cola y esta selección se respeta para todos los mensajes transferidos a la misma cola utilizando el manejador de objeto obtenido.

Al especificar `MQOO_BIND_ON_OPEN`, fuerza a que todos los mensajes se direccionen al mismo destino. Por lo tanto, las aplicaciones con afinidades de mensajes no se ven afectadas. Si el destino no está disponible, los mensajes permanecen en la cola de transmisión hasta que éste vuelve a estar disponible.

`MQOO_BIND_ON_OPEN` también se aplica cuando el nombre del gestor de colas se especifica en el descriptor de objeto al abrir una cola. Puede haber más de una ruta al gestor de colas especificado. Por ejemplo, puede haber varias rutas de red u otro gestor de colas puede haber definido un alias. Si especifica `MQOO_BIND_ON_OPEN`, se selecciona una ruta cuando se abre la cola.

Nota: Esta es la técnica recomendada. No obstante, no funciona en una configuración multisalto en la que un gestor de colas anuncia un alias para una cola de clúster. Tampoco ayuda en situaciones en las que las aplicaciones utilizan colas diferentes en el mismo gestor de colas para diferentes grupos de mensajes.

Una alternativa a la especificación de `MQOO_BIND_ON_OPEN` en la llamada `MQOPEN` es modificar las definiciones de cola. En las definiciones de cola, especifique `DEFBIND(OPEN)` y permita que la opción `DefBind` de la llamada `MQOPEN` tome como valor predeterminado `MQOO_BIND_AS_Q_DEF`.

Establecer la opción MQOO_BIND_ON_GROUP en la llamada MQOPEN

Fuerce a que todos los mensajes de un grupo se coloquen en el mismo destino utilizando la opción `MQOO_BIND_ON_GROUP` en la llamada `MQOPEN`. Se debe especificar `MQOO_BIND_ON_OPEN` o `MQOO_BIND_ON_GROUP` cuando se utilizan grupos de mensajes con clústeres para asegurarse de que todos los mensajes del grupo se procesan en el mismo destino.

Al abrir una cola y especificar `MQOO_BIND_ON_GROUP`, fuerza a que todos los mensajes de un grupo que se envían a esta cola se envíen a la misma instancia de la cola. `MQOO_BIND_ON_GROUP` enlaza todos los

mensajes de un grupo con el mismo gestor de colas, y también con la misma ruta. Por ejemplo, si hay una ruta IP y una ruta NetBIOS al mismo destino, se selecciona una de ellas cuando se abre la cola y esta selección se respeta para todos los mensajes de un grupo transferidos a la misma cola utilizando el manejador de objeto obtenido.

Al especificar MQOO_BIND_ON_GROUP, fuerza a que todos los mensajes de un grupo se direccionen al mismo destino. Por lo tanto, las aplicaciones con afinidades de mensajes no se ven afectadas. Si el destino no está disponible, los mensajes permanecen en la cola de transmisión hasta que éste vuelve a estar disponible.

MQOO_BIND_ON_GROUP también se aplica cuando el nombre del gestor de colas se especifica en el descriptor de objeto al abrir una cola. Puede haber más de una ruta al gestor de colas especificado. Por ejemplo, puede haber varias rutas de red u otro gestor de colas puede haber definido un alias. Si especifica MQOO_BIND_ON_GROUP, se selecciona una ruta cuando se abre la cola.

Para que MQOO_BIND_ON_GROUP sea efectivo, debe incluir la opción put MQPMO_LOGICAL_ORDER en MQPUT. Puede establecer **GroupId** en el MQMD del mensaje en MQGI_NONE y debe incluir los distintivos de mensaje siguientes en el campo **MsgFlags** del MQMD de los mensajes:

- Último mensaje en grupo: MQMF_LAST_MSG_IN_GROUP
- Todos los otros mensajes en grupo: MQMF_MSG_IN_GROUP

Si se especifica MQOO_BIND_ON_GROUP pero los mensajes no están agrupados, el comportamiento es similar al de MQOO_BIND_NOT_FIXED.

Nota: Esta es la técnica recomendada para garantizar que los mensajes de un grupo se envían al mismo destino. Sin embargo, no funciona en una configuración de salto por múltiples sitios en la que un gestor de colas anuncia un alias para una cola de clúster.

Una alternativa a la especificación de MQOO_BIND_ON_GROUP en la llamada MQOPEN es modificar las definiciones de cola. En las definiciones de cola, especifique DEFBIND(GROUP) y permita que la opción DefBind de la llamada MQOPEN tome como valor predeterminado MQOO_BIND_AS_Q_DEF.

Escribir un programa de salida de carga de trabajo de clúster personalizado

En lugar de modificar las aplicaciones, puede eludir el problema de las afinidades de mensajes escribiendo un programa de salida de carga de trabajo de clúster. Escribir un programa de salida de carga de trabajo de clúster no es fácil y no es una solución recomendada. El programa tendría diseñarse para reconocer la afinidad inspeccionando el contenido de los mensajes. Una vez reconocida la afinidad, el programa tendría que forzar al programa de utilidad de gestión de carga de trabajo a direccionar todos los mensajes relacionados al mismo gestor de colas.

Multi Configuración de un clúster uniforme

Los clústeres uniformes permiten que se diseñen las aplicaciones para la escala y la disponibilidad, y se puedan conectar a cualquiera de los gestores de colas dentro de ese clúster uniforme.

Antes de empezar

Para obtener una introducción a la agrupación en clúster, consulte Clústeres. Para obtener una introducción a los clústeres uniformes, consulte “Acerca de los clústeres uniformes” en la página 432.

Acerca de esta tarea

Los clústeres uniformes utilizan la agrupación en clúster de IBM MQ para la comunicación entre los gestores de colas y el equilibrio de carga de trabajo entre colas. Sin embargo, difieren de los clústeres típicos de IBM MQ de las formas siguientes:

- Los clústeres uniformes normalmente tienen un número menor de gestores de colas en el clúster. No debe crear un clúster uniforme con más de 10 gestores de colas.
- Cada miembro del clúster tiene una configuración casi idéntica.

- Normalmente, el clúster lo utiliza una sola aplicación o un grupo de aplicaciones relacionadas.
- El número de instancias de aplicación que se conectan al clúster debe ser mayor que, o igual a, el número de gestores de colas.

Puede simplificar la creación de un clúster uniforme y, posteriormente, hacer que la configuración de los miembros del clúster uniforme sea idéntica, utilizando el soporte de configuración automática y agrupación en clúster automática.

Procedimiento

- Más información sobre clústeres uniformes
- Crear un clúster uniforme
- Crear un clúster uniforme
- Suspender un gestor de colas de un clúster uniforme

Multi Acerca de los clústeres uniformes

El objetivo de un despliegue de clúster uniforme es que las aplicaciones se puedan diseñar para la escala y la disponibilidad, y que puedan conectarse a cualquiera de los gestores de colas dentro del clúster uniforme. Esto elimina cualquier dependencia de un gestor de colas específico, lo que produce una mejor disponibilidad y equilibrio de carga de trabajo del tráfico de mensajería. z/OS Los clústeres uniformes no están disponibles en IBM MQ for z/OS, los grupos de compartición de colas proporcionan muchas de las prestaciones de un clúster uniforme.

Los clústeres uniformes son un patrón específico de un clúster de IBM MQ que proporciona una pequeña colección de gestores de colas de alta disponibilidad y escalado horizontal. Estos gestores de colas se configuran de forma casi idéntica, de modo que una aplicación puede interactuar con ellos como un único grupo. Esto facilita la tarea de asegurarse de que se utilice cada uno de los gestores de colas del clúster, garantizando automáticamente que las instancias de aplicación se distribuyen uniformemente entre los gestores de colas.

Los clústeres uniformes eliminan algunos de los pasos manuales que un administrador debe realizar para crear y administrar un grupo de gestores de colas independientes interconectados. Mueven alguna lógica de conexión de cliente desde el cliente al gestor de colas, donde la información sobre los niveles de actividad de la aplicación puede conformar las decisiones en los clientes, en cuanto a qué gestores de colas deben conectarse.

Puede simplificar la creación inicial de un clúster uniforme y, posteriormente, hacer que la configuración de los miembros del clúster uniforme sea idéntica, utilizando el soporte de configuración automática y agrupación en clúster automática. Cuando se utiliza esta capacidad, un archivo de configuración describe el clúster y otro representa la configuración MQSC para que se aplicará a todos los gestores de colas del clúster uniforme. Cada vez que se reinicie un gestor de colas, la configuración se aplicará de nuevo y se formará automáticamente el clúster. Consulte “Creación de un clúster uniforme” en la página 447 para obtener más detalles sobre la utilización de esta característica.

Para aprovechar al máximo un clúster uniforme, también se debe escalar cada aplicación en varias instancias coincidentes, preferiblemente con al menos tantas instancias como gestores de colas, si no muchas más.

Un clúster de IBM MQ, sea cual sea el tamaño, proporciona varias prestaciones:

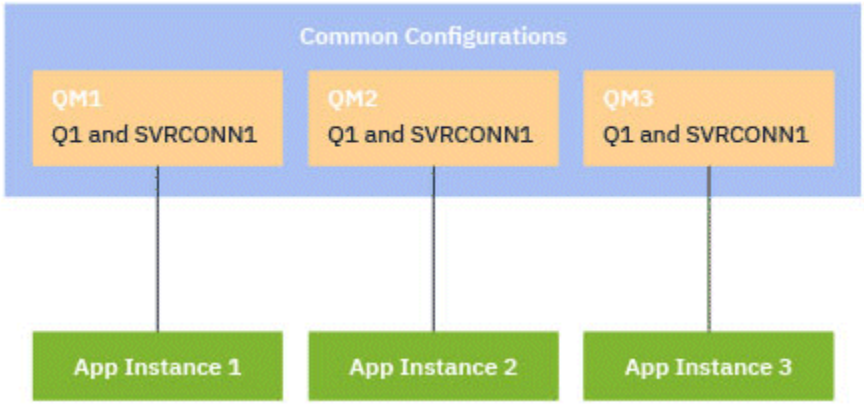
- Un directorio de todos los recursos de clúster, que puede ser descubierto por cualquier miembro de un clúster
- Creación automática de canales y conectividad
- Escalado horizontal entre varias colas coincidentes utilizando el equilibrio de carga de trabajo de mensajes
- Direccionamiento de mensajes dinámico según la disponibilidad

Los clústeres uniformes utilizan la agrupación en clúster de IBM MQ para la comunicación entre los gestores de colas y el equilibrio de carga de trabajo entre colas. Sin embargo, difieren de los clústeres típicos de IBM MQ de las formas siguientes:

- Los clústeres uniformes normalmente tienen un número menor de gestores de colas en el clúster. No debe crear un clúster uniforme con más de 10 gestores de colas.
- Cada miembro del clúster tiene una configuración casi idéntica.
- Normalmente, el clúster lo utiliza una sola aplicación o un grupo de aplicaciones relacionadas.
- El número de instancias de aplicación que se conectan al clúster debe ser mayor que, o igual a, el número de gestores de colas.

En un patrón de clúster uniforme, todos los gestores de colas del clúster ofrecen los mismos servicios de mensajería. Por ejemplo, puede configurar todos los miembros del clúster para que tengan las mismas colas locales definidas, y permitir que las aplicaciones cliente se conecten a cualquier miembro del clúster. Podría también tener los mismos canales de conexión de servidor definidos y posiblemente los mismos registros de autorización, las mismas reglas de autenticación de canal, etc. Sin embargo, todavía es posible que los miembros del clúster tengan algunas diferencias en los objetos y en la configuración. Por ejemplo, algunas aplicaciones pueden crear colas dinámicas temporales cuando están conectadas a un gestor de colas. Además, algunas actualizaciones de configuración se podrían implantar en los miembros durante un período de tiempo; por ejemplo, los certificados nuevos o actualizados. Al igual que con los clústeres de IBM MQ regulares, dos de los gestores de colas requerirán una configuración adicional para que sean gestores de colas de repositorio completos.

El diagrama siguiente muestra que los gestores de colas tienen configuraciones similares. Definen la misma cola denominada Q1 y el mismo canal de conexión de servidor SVRCONN1.



Tenga en cuenta que para que varios gestores de colas con nombres de canal de conexión de servidor idénticos funcionen con una única tabla de definición de canal de cliente (CCDT), debe utilizar el formato de CCDT actualizado introducido en IBM MQ 9.1.2. Consulte “Configuración de una tabla de definición de canal de cliente en formato JSON” en la página 47.

Nombres de aplicación e instancias de aplicación

Un nombre de aplicación se muestra como el atributo APPLTAG del mandato **DISPLAY CONN(*) TYPE CONN**. A partir de IBM MQ 9.1.2, también hay un cambio en el modo en que se establece el nombre de aplicación.

Una instancia de una aplicación es un conjunto de conexiones estrechamente relacionadas que proporcionan una *unidad de ejecución* para dicha aplicación. Normalmente, se trata de un único proceso de sistema operativo, que puede tener distintas hebras y conexiones IBM MQ asociadas.

Para obtener más información sobre el nombre de aplicación y las instancias de aplicación, consulte Conceptos de desarrollo de aplicaciones.

Clientes que se pueden volver a conectar

Los clientes reconectables pueden moverse para conseguir una distribución de carga de trabajo uniforme mientas que, por definición, un cliente no reconectable no puede reconectarse a un gestor de colas distinto. Sin embargo, todavía puede haber una buena razón para conectar un cliente no reconectable a un clúster uniforme: por ejemplo, porque el cliente crea algún tipo de estado persistente, y se utiliza algún otro mecanismo para asegurarse de que hay instancias de la aplicación que se ejecutan en cada uno de los gestores de colas.

Aplicaciones enlazadas localmente

Se espera que los clústeres uniformes tengan aplicaciones de IBM MQ que se conecten como aplicaciones cliente, en lugar de aplicaciones enlazadas localmente. No se impide la conexión de las aplicaciones enlazadas a miembros de clúster uniformes, pero con las aplicaciones enlazadas localmente los clústeres uniformes no pueden lograr una distribución de carga de trabajo uniforme, porque estas no pueden conectarse a ningún otro miembro del clúster.

Tareas relacionadas

Especificación del nombre de aplicación en los lenguajes de programación admitidos

Multi *Equilibrio de aplicaciones automático*

El equilibrado automático de aplicaciones mejora considerablemente la distribución y la disponibilidad de las aplicaciones habilitando un clúster uniforme de IBM MQ para gestionar de cerca la distribución de aplicaciones en todo el clúster, en lugar de depender de la aleatorización o de un anclaje manual de aplicaciones a gestores de colas específicos.

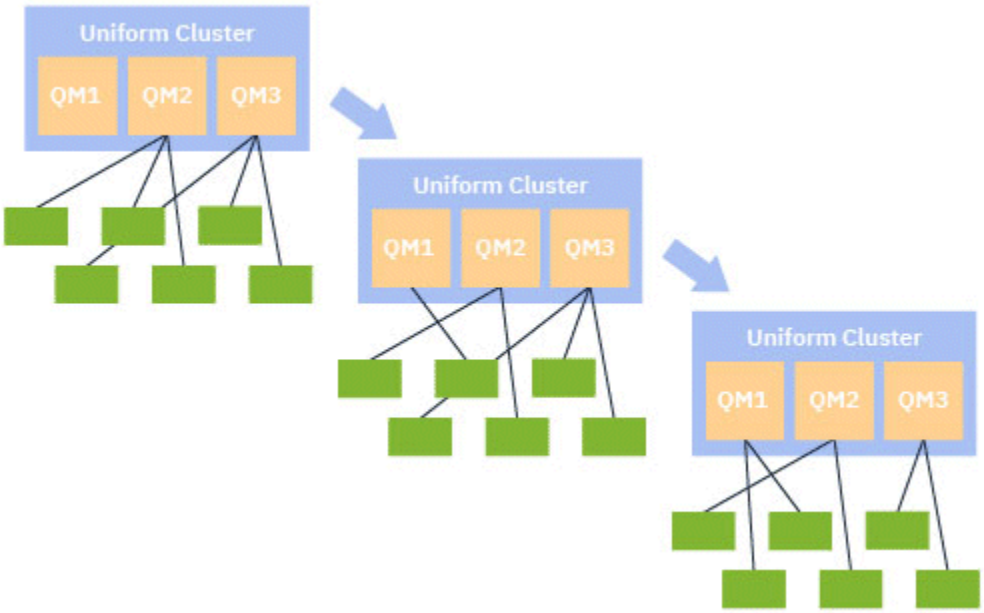
El equilibrio automático entre un conjunto de gestores de colas en clúster está soportado para aplicaciones escritas en C, JMS, IBM MQ .NET, XMS .NET.

Cuando hay al menos tantas instancias de la misma aplicación que gestores de colas, el clúster uniforme garantiza constantemente que cada gestor de colas tenga al menos una instancia de la aplicación conectada.

Las aplicaciones pueden eliminar una afinidad específica a un gestor de colas y, en su lugar, utilizar una tabla de definiciones de canal de cliente (CCDT) para aleatorizar la conectividad con el grupo de gestores de colas del clúster uniforme de forma segura. Las aplicaciones pueden hacer esto por los siguientes motivos:

- Cuando hay suficientes instancias de aplicación consumidora, siempre hay una instancia de la aplicación que procesa los mensajes.
- Cuando se detiene un gestor de colas, las instancias de aplicación conectadas se distribuyen uniformemente entre los gestores de colas restantes del clúster.
- Cuando inicia un gestor de colas, todas las instancias de aplicación conectadas a otros gestores de colas del clúster se reequilibran automáticamente para incluir el gestor de colas que se acaba de iniciar.

Esto significa que el clúster uniforme continuamente garantiza que las aplicaciones se distribuyen de forma óptima, con lo que se maximiza el proceso de mensajes, incluso en el caso de paradas planeadas y no planeadas.



Para lograr el equilibrio automático, los gestores de colas en el clúster uniforme periódicamente comparten información entre ellos. Lo hacen publicando metadatos en temas del sistema bajo la rama $SYS/MQ reservada del árbol de temas. Cada gestor de colas del clúster uniforme se suscribe a los mensajes publicados por otros gestores de colas y crea una imagen del estado de las aplicaciones en el clúster uniforme.

Los gestores de colas supervisan la distribución de las aplicaciones cliente en todo el clúster. Cuando el número de aplicaciones conectadas a un gestor de colas específico es lo suficientemente bajo para determinar que el clúster está desequilibrado, dicho gestor de colas publica una solicitud en un tema del sistema a uno de los demás gestores de colas del clúster.

Cuando se recibe el mensaje, el gestor de colas de destino solicita a una de sus aplicaciones cliente que se redirija al gestor de colas solicitante. La aplicación cliente recibe la solicitud de redirección, cierra su conexión y se vuelve a conectar al gestor de colas solicitante. Este mecanismo de equilibrio automático es transparente para la aplicación. Para obtener más información, consulte “Cómo funciona el equilibrio automático” en la página 436.

Mediante la distribución periódica de metadatos a las aplicaciones conectadas, el clúster uniforme puede lograr una proporción equilibrada de las aplicaciones cliente en los gestores de colas a lo largo del tiempo. Para impedir que se produzcan sucesos de redirección rápida sucesiva, el algoritmo de equilibrio automático limita la velocidad a la que se realizan las solicitudes de redirección.

Puede supervisar el estado actual de las aplicaciones en todos los gestores de colas de un clúster y supervisar las instancias de aplicación. Para obtener más información, consulte Supervisión del equilibrio de aplicaciones. También puede resolver diversos problemas con el equilibro de aplicaciones, tal como se describe en Resolución de problemas de equilibrio de aplicaciones.

El reequilibrio solo es útil para las aplicaciones con un tiempo de conexión largo. Si tiene aplicaciones cliente con tiempos de conexión cortos, por ejemplo, las aplicaciones cliente que se escriben para conectarse y desconectarse con regularidad a distintos gestores de colas, debería configurarlas como no reconectables. Esto las elimina del conjunto de aplicaciones que los gestores de colas intentan equilibrar.

Conceptos relacionados

“Cómo el equilibrado automático utiliza la reconexión automática” en la página 438
El equilibrio automático de clúster uniforme utiliza mejoras en la característica de reconexión automática existente de IBM MQ.

Multi *Cómo funciona el equilibrio automático*

En el clúster uniforme, las conexiones de cliente se agrupan juntas basándose en el nombre de aplicación. Las aplicaciones que se conectan a cualquier miembro del clúster uniforme utilizando el mismo nombre de aplicación se consideran equivalentes a cualquier otra aplicación que utilice el mismo nombre de aplicación.

El equilibrio automático garantiza una extensión uniforme de las instancias de aplicación entre los miembros del clúster, consulte “Nombres de aplicación e instancias de aplicación” en la página 433 para obtener más información. Utilice el mandato DISPLAY APSTATUS para visualizar el estado de una o más aplicaciones e instancias de aplicación, conectadas a un gestor de colas o a un clúster uniforme.

Por ejemplo, puede establecer que todas las instancias de una aplicación de solicitud de seguro tengan un nombre de aplicación de "INSURANCE.REQUESTS". Las conexiones relacionadas desde esta aplicación se agruparán automáticamente en instancias, según corresponda, y el equilibrio se realizará individualmente para cada una de ellas.

Cuando las nuevas instancias de la aplicación se conectan a un miembro del clúster uniforme, el algoritmo de equilibrio automático evalúa qué gestores de colas tienen el menor número de instancias de INSURANCE.REQUESTS y redirige algunas conexiones a esos gestores de colas.

El equilibrio automático solo se habilita en las circunstancias siguientes:

- El valor SHARECNV del canal es mayor que cero.
- Una de las condiciones siguientes es cierta:
  - La aplicación cliente especifica MQCNO_RECONNECT
  - El archivo `mqclient.ini` especifica **`Defrecon`**=YES

Nota: Las aplicaciones con afinidad de gestor de colas, por ejemplo, debido a una suscripción duradera o a una cola de respuesta dinámica, no se pueden reequilibrar de forma segura y deben utilizar MQCNO_RECONNECT_QMGR o ninguna opción de reconexión en absoluto.

Cuando un cliente se redirige a un gestor de colas alternativo, como de costumbre utilizará las tablas de definición de canal de cliente local (CCDT) para localizar la información de conexión para el nuevo destino. Por lo tanto, es importante que el funcionamiento del equilibrio automático sea fluido y eficiente, que los clientes utilicen una CCDT que contenga una entrada para cada miembro del clúster uniforme, así como cualquier grupo de gestores de colas utilizado para equilibrar las conexiones iniciales.

El uso de una CCDT en formato JSON simplifica esto, ya que permite que varias conexiones utilicen el mismo nombre de conexión de servidor. Para obtener más información, consulte “Configuración de una tabla de definición de canal de cliente en formato JSON” en la página 47.

Conceptos relacionados

“Cómo el equilibrado automático utiliza la reconexión automática” en la página 438
El equilibrio automático de clúster uniforme utiliza mejoras en la característica de reconexión automática existente de IBM MQ.

ALW *Equilibrio automático de aplicaciones JMS*

Cuando las aplicaciones Jakarta Messaging 3.0 o Java Message Service 2.0 se equilibran automáticamente, los grupos subyacentes de conexiones IBM MQ que crean las aplicaciones JMS se mueven juntos.

A partir de IBM MQ 9.3.0, la propiedad **`dynamicallyBalanced`** está disponible al configurar ActivationSpecs. Esta propiedad especifica si se puede solicitar a un MDB que reciba mensajes de un gestor de colas diferente como parte del equilibrio de aplicaciones en un clúster uniforme. Para obtener más información, consulte Configuración del adaptador de recursos para la comunicación de entrada.

Para la gestión de las conexiones JMS, los clústeres uniformes tienen el concepto de una *instancia de aplicación*. Para JMS, una *instancia de aplicación* se define como una conexión JMS y sus sesiones JMS asociadas.

Se asigna una etiqueta de conexión exclusiva en la conexión de cliente correspondiente a la conexión JMS, y a continuación se aplica la misma etiqueta a las conexiones de cliente correspondientes a las sesiones JMS que crea esta conexión JMS.

Por ejemplo, si un par de aplicaciones cliente ejecutan aplicaciones JMS en un clúster uniforme con un único gestor de colas activo (Gestor de colas 1):

- El cliente 1 crea una fábrica de conexiones en la que establece un nombre de aplicación de "App1", y crea una conexión JMS y tres sesiones JMS. El cliente 1 crea cuatro conexiones de cliente en el Gestor de colas 1, donde cada una de ellas comparte el mismo código de conexión, y esto se trata como una única instancia de "App1".
- El cliente 2 también crea una fábrica de conexiones en la que establece un nombre de aplicación de "App1", y crea una conexión JMS y dos sesiones JMS. El cliente 2 crea tres conexiones de cliente, cada una de las cuales comparte la misma etiqueta de conexión (distinta de la asignada al cliente 1), y esto se trata como una única instancia distinta de "App1".
- Por lo tanto, el gestor de colas ve dos instancias de "App1".

Cuando se realiza el equilibrio automático, las instancias de aplicación se mueven. Un gestor de colas elige una instancia de aplicación (un grupo de conexiones de cliente que comparten la misma etiqueta de conexión) y solicita que la instancia se mueva a un gestor de colas distinto. El código de cliente recibe la solicitud y se asegura de que todas las conexiones relacionadas (correspondientes a una conexión JMS y sus sesiones JMS asociadas) se muevan al nuevo gestor de colas.

Por ejemplo, tome el conjunto de instancias de aplicación descritas anteriormente, y suponga que un nuevo gestor de colas (Gestor de colas 2) se inicia en el clúster uniforme.

El gestor de colas 2 no tiene trabajo, pero el gestor de colas 1 tiene 2 instancias de "App1", por lo que el gestor de colas 2 solicita que el gestor de colas 1 transfiera una instancia de "App1" al gestor de colas 2.

El gestor de colas 1 elige una instancia de "App1" para mover. A los efectos del ejemplo, supongamos que elige la instancia creada por el cliente 1.

- El gestor de colas 1 envía una solicitud al cliente 1 para mover su instancia de "App1" a QM2.
- El cliente cierra sus cuatro conexiones de cliente existentes con el gestor de colas 1 y crea cuatro nuevas conexiones con el gestor de colas 2.
- La conexión JMS y sus sesiones JMS, excepto durante una breve pausa en el proceso, normalmente no deberían resultar alteradas.

Nota:

Una aplicación podría recibir una excepción JMS si determinadas operaciones están en curso en el momento en que se mueve una instancia de aplicación.

La excepción JMS tendrá una excepción IBM MQ vinculada, de la que se puede recuperar el código de razón para determinar la causa de la anomalía.

Los códigos de razón esperados son los siguientes:

MQRC_CALL_INTERRUPTED
: Se produce cuando, por ejemplo, una reconexión interrumpe una operación de poner un mensaje persistente (el valor predeterminado en JMS) fuera de un punto de sincronización.

MQRC_BACKED_OUT
: Se produce cuando, por ejemplo, una reconexión interrumpe un intento de poner un mensaje dentro de un punto de sincronización.

Conceptos relacionados

“Cómo funciona el equilibrio automático” en la página 436
En el clúster uniforme, las conexiones de cliente se agrupan juntas basándose en el nombre de aplicación. Las aplicaciones que se conectan a cualquier miembro del clúster uniforme utilizando el mismo nombre de aplicación se consideran equivalentes a cualquier otra aplicación que utilice el mismo nombre de aplicación.

“Cómo el equilibrado automático utiliza la reconexión automática” en la página 438

El equilibrio automático de clúster uniforme utiliza mejoras en la característica de reconexión automática existente de IBM MQ.

Multi *Cómo el equilibrado automático utiliza la reconexión automática*

El equilibrio automático de clúster uniforme utiliza mejoras en la característica de reconexión automática existente de IBM MQ.

En las versiones de IBM MQ anteriores a IBM MQ 9.2.0, la característica de reconexión automática se reconecta automáticamente a una instancia en espera de un gestor de colas o a un gestor de colas diferente, basándose en los detalles de conexión proporcionados, normalmente una lista de nombres de conexión o una tabla de definición de canal de cliente (CCDT).

En algunas circunstancias, el cliente de IBM MQ realiza de forma silenciosa la reconexión sin que la aplicación sepa que se ha producido. La decisión de qué se reconectará al gestor de colas depende totalmente de la secuencia de nombres de conexión en una lista de nombres de conexión o de la configuración de equilibrio de carga de trabajo en la tabla de definición de canal de cliente.

Desde IBM MQ 9.2.0 se puede enviar una solicitud de reconexión a un cliente que contiene una sugerencia de a qué gestor de colas debe reconectarse el cliente. En muchos casos de ejemplo de reconexión, como una anomalía del gestor de colas, o el administrador que emite el mandato **endmqm -r**, no se incluye un nombre de gestor de colas en la información de sugerencia, y el comportamiento de reconexión automática funciona como lo hacía antes IBM MQ 9.2.0.

Sin embargo, si ha configurado un clúster uniforme, el equilibrado de aplicaciones automático envía periódicamente solicitudes de reconexión a los clientes, para poder lograr un clúster equilibrado. En estos casos, el clúster uniforme especifica un nombre de gestor de colas en la sugerencia de reconexión para asegurarse de que las conexiones de cliente se mueven a los gestores de colas que menos conexiones tienen.

Para que el equilibrio automático funcione, deben estar en vigor los valores siguientes:

- Las aplicaciones de IBM MQ utilicen tablas de definición de canal de cliente para recuperar de ellas la información de conexión.
- Las CDT contienen una entrada para cada gestor de colas del clúster uniforme.

Si este no es el caso, el clúster no puede equilibrar automáticamente las aplicaciones entre todos los miembros del clúster.

Si una aplicación está utilizando una versión del cliente de IBM MQ anterior a IBM MQ 9.2.0 y está configurada para dar soporte a la reconexión de cliente automática, es posible que el clúster uniforme envíe una solicitud para que lleve a cabo los pasos de reconexión.

No se pedirá al cliente que se reconecte a un gestor de colas específico, sino que realice la misma secuencia de lógica de reconexión que haría para otros sucesos de reconexión. Para lograr una distribución uniforme de las aplicaciones cliente anteriores a IBM MQ 9.2.0 en el clúster uniforme, asegúrese de que los clientes estén configurados para utilizar CCDT que contengan entradas ponderadas de forma uniforme para cada miembro del clúster.

Las aplicaciones pueden realizar varios intentos de reconexión antes de conectarse a un gestor de colas que necesita la instancia adicional, por lo que es una forma menos eficiente de lograr una distribución uniforme de las aplicaciones en el clúster. El equilibrado automático puede llevar más tiempo en alcanzarse en estos entornos.

Los clientes de IBM MQ no admiten la reconexión de cliente automática

Si una aplicación está utilizando una versión del cliente de IBM MQ que no admite a la reconexión de cliente automática, es posible que la aplicación reciba un código de retorno de anomalía de una llamada de MQI.

Si la aplicación no se ha diseñado para gestionar las anomalías y realizar manualmente las reconexiones, es posible que sea necesario inhabilitar el equilibrado automático para esas aplicaciones.

Nota: el equilibrado automático está habilitado para cualquier aplicación que se ha identificado como reconectable, es decir, la aplicación tiene MQCNO_RECONNECT en sus opciones de conexión efectiva.

Tareas relacionadas

“Creación de un nuevo clúster uniforme” en la página 448
Cómo crear un nuevo clúster uniforme.

Multi *Influir en el reequilibrio de aplicaciones en clústeres uniformes*

Con el equilibrio automático de aplicaciones (una característica de clústeres uniformes), se puede solicitar a una conexión de aplicación que se mueva a un gestor de colas alternativo en cualquier momento de su ciclo de vida.

Introducción

A partir de IBM MQ 9.3.0, el algoritmo de equilibrio intenta automáticamente tener en cuenta el estado de las aplicaciones para minimizar la interrupción del flujo de aplicaciones. Esto se puede ajustar para adaptarse a aplicaciones o instancias de aplicaciones específicas proporcionando a IBM MQ información adicional sobre el tipo de aplicación o el patrón de actividad de IBM MQ que realiza esta aplicación.

Por lo general, es probable que la persona que desarrolla o despliega una aplicación cliente sea la mejor indicada para comprender este patrón y proporcionar esta información al gestor de colas (consulte Despliegue de aplicaciones de cliente flexibles y escalables), pero también puede, o además, ser ajustado por un administrador.

Tenga en cuenta que si el gestor de colas no puede lograr una distribución uniforme de las aplicaciones en un periodo de tiempo razonable, es posible que las conexiones de aplicación se vuelvan a equilibrar con otros gestores de colas sin esperar un momento adecuado en el flujo de IBM MQ.

Esto también se puede ajustar para cumplir con los requisitos. Si es más importante conseguir rápidamente una distribución uniforme de las aplicaciones, puede configurar el producto para que espere menos tiempo para encontrar un momento adecuado para reequilibrar una aplicación. De forma alternativa, si es más importante evitar la interrupción de las aplicaciones, es posible configurar el producto para que siempre espere un momento adecuado para mover la aplicación.

Consulte Despliegue de aplicaciones de cliente flexibles y escalables para obtener más información sobre la visión general.

Para aplicaciones .NET, consulte “Influir en el reequilibrio de aplicaciones en .NET” en la página 442 para obtener más información.

Para aplicaciones .XMS.NET, consulte Propiedades de ConnectionFactory para obtener más información.

V 9.4.0 Para Aplicaciones JMS, consulte “Influir en el reequilibrio de aplicaciones en IBM MQ classes for JMS” en la página 443 para obtener más información.

Comportamiento predeterminado de equilibrio de aplicaciones

De forma predeterminada, la transacción/unidad de estado de trabajo de una interacción de aplicaciones con un gestor de colas se considera para todas las aplicaciones.

Para transacciones locales, el equilibrio automático de aplicaciones evita emitir solicitudes de reequilibrio a aplicaciones que actualmente están implicadas en una transacción. Si bien esto no elimina la posibilidad de que una aplicación reciba un código de retorno de copia de seguridad, ya que alcanzar el tiempo de espera de reequilibrio configurado o una interrupción real podría causar un código de retorno de este tipo, significa que normalmente no se solicitará a las aplicaciones que se vuelvan a conectar mientras estén en medio de una transacción.

Para las aplicaciones que inician una nueva transacción casi inmediatamente después de completar la anterior, puede haber un retardo para la llamada inicial en la nueva transacción mientras se completa el reequilibrio. Esto asegura que el equilibrio automático de aplicaciones todavía puede lograr una distribución uniforme de las aplicaciones entre los gestores de colas en un clúster uniforme.

Si tiene aplicaciones que utilizan transacciones de ejecución más larga, es posible que desee considerar la posibilidad de aumentar el valor del tiempo de espera de reequilibrio o inhabilitar esta restricción por completo. Consulte “Configuración del comportamiento de equilibrio” en la página 441 para obtener enlaces sobre cómo controlar esto en MQI y .NET, o 'Diseño de aplicaciones cliente para tolerancia a errores y escalabilidad' para el equivalente de nivel de código.

Equilibrio solicitud-respuesta

Cuando el tipo de aplicación se especifica como **Request-Reply**, se espera una respuesta GET para cada PUT que realiza la instancia de aplicación. Si la instancia de aplicación implica varias hebras o se ocupa de solicitudes y respuestas en lotes, varias solicitudes y respuestas pueden estar en curso en un momento dado.

La aplicación no se considera apta para moverse, hasta que el número de solicitudes enviadas sea igual al número de respuestas recibidas, o hasta que se supere el valor de retroceso del tiempo de espera.

Una excepción a esto es cuando la caducidad del mensaje está configurada para un mensaje de solicitud. Se supone que las respuestas deben recibirse dentro del intervalo de caducidad del mensaje de solicitud y cuando todos los mensajes de solicitud han caducado, el algoritmo de equilibrio ya no espera respuestas adicionales antes de considerar la instancia elegible para moverse.

Si hay varias solicitudes pendientes, sólo se considera la última caducidad entre los mensajes de solicitud enviados. Cuando se utilizan valores de caducidad significativos, debe configurar el parámetro de equilibrio de **Timeout** para que la aplicación sea un valor como mínimo tan alto como cualquier caducidad de mensaje enviado, para evitar reducir cualquier intervalo de caducidad de solicitud/respuesta esperada.

El patrón anterior sólo es adecuado para aplicaciones que esperan tener periodos en los que no hay solicitudes pendientes. Las aplicaciones multihebra complejas, que envían y reciben mensajes constantemente, por ejemplo, nunca pueden ser elegibles para el reequilibrio bajo este patrón.

Notas:

- No se realiza ningún intento de correlacionar solicitudes y respuestas específicas, por lo que, si caduca una respuesta anterior dentro de un lote de mensajes en curso, la aplicación puede seguir esperando hasta que caduque la última solicitud antes de ser elegible para el equilibrio.
- En particular, es necesario tener cuidado si se combina un tiempo de caducidad ilimitado y mensajes que caducan, por razones similares.

  Si los mensajes de solicitud con una caducidad limitada están pendientes, y los mensajes nuevos se envían con un tiempo de caducidad ilimitado, el tiempo de espera de caducidad ilimitado *no* se tiene en cuenta por el algoritmo de equilibrio, que sigue cumpliendo la hora de caducidad actual más reciente.

  De lo contrario, las respuestas anteriores que hayan caducado podrían impedir que la aplicación se pudiera mover. En consecuencia, si las respuestas de tiempo de espera de caducidad ilimitadas están pendientes, pero las solicitudes que caducan se envían posteriormente, el tiempo de espera se reduce a la caducidad más larga (limitada).

  En general, debe evitar que una sola instancia de aplicación envíe mensajes de solicitud tanto caducados como no caducados en una aplicación equilibrada, ya que la elegibilidad para reequilibrar resulta más difícil para un desarrollador o administrador para realizar un seguimiento o definir con precisión.
- Sólo la hora de caducidad especificada por la aplicación emisora (por ejemplo, en la MQI el valor de MQMD.**Expiry**) se tiene en cuenta al determinar el tiempo de espera de las respuestas. Las modificaciones posteriores de este valor, por ejemplo, la utilización de CAPEXPRY no afectarán al tiempo de espera.

Configuración del comportamiento de equilibrio

Para influir con precisión cuando IBM MQ vuelve a equilibrar las aplicaciones, determinados entornos de aplicaciones cliente pueden proporcionar información en tiempo de conexión sobre el patrón de mensajería que se está utilizando.

Esta información se proporciona en una nueva estructura a la que se hace referencia como *Opciones de equilibrio*.

Para la MQI, consulte "Configuración del comportamiento de equilibrio utilizando la MQI" en la página 441.

Para el equivalente de cliente .NET de esta estructura, consulte "Influir en el reequilibrio de aplicaciones en .NET" en la página 442.

V 9.4.0 Para que el enfoque JMS establezca estas opciones, consulte "Influir en el reequilibrio de aplicaciones en IBM MQ classes for JMS" en la página 443 para obtener más información.

Otros entornos de cliente no soportan actualmente el suministro de esta estructura en el momento de la conexión.

Multi *Configuración del comportamiento de equilibrio utilizando la MQI*

Para influir con precisión cuando IBM MQ vuelve a equilibrar las aplicaciones, determinados entornos de aplicaciones cliente pueden proporcionar información en tiempo de conexión sobre el patrón de mensajería que se está utilizando.

En la MQI, la estructura de opciones de equilibrio se conoce como MQBNO.

Si no se proporciona ninguna *Opción de equilibrio* en el programa, los clientes de soporte obtienen esta información en la stanza Application o la stanza applicationDefaults en el archivo `client.ini` desplegado junto a la aplicación cliente.

Nota: Estas stanzas son idénticas, excepto que la versión `Application` contiene un campo **Name** para identificar a qué aplicación se aplican estas opciones.

Si se proporciona cualquiera de las formas de stanza, se requiere que todos los campos estén presentes, excepto **BalanceOptions**, que se supone que es none si no se establece explícitamente.

El orden de preferencia para el suministro de opciones es:

1. La aplicación en CONNX proporciona una estructura MQBNO y se utiliza en su totalidad
2. O bien, la stanza `Application` coincidente especifica, si está presente, sólo se utiliza para generar una
3. O bien, la stanza `ApplicationDefaults`, si está presente, sólo se utiliza para generar una
4. O bien, no hay flujos de MQBNO para esta conexión.

Puede proporcionar tres partes de información clave de la estructura MQBNO o del archivo `client.ini`:

1. **ApplicationType** o el patrón de aplicación.

   Este campo indica a IBM MQ el patrón general de actividad de IBM MQ en el que participa esta aplicación.

   Se da soporte a tres tipos de aplicaciones:

   Sencillo
   No se deben aplicar reglas específicas más allá de los valores predeterminados que se describen en "Comportamiento predeterminado de equilibrio de aplicaciones" en la página 439.

   Solicitud-Respuesta
   Después de cada llamada MQPUT, se espera una llamada MQGET coincidente para un mensaje de respuesta. Consulte "Equilibrio solicitud-respuesta" en la página 440 para obtener más detalles.

Cliente gestionado
Las solicitudes de reequilibrio siempre se envían inmediatamente al cliente, lo que se reequilibra en un punto que considere apropiado, por ejemplo, el adaptador de recursos JEE se registraría de esta manera.

2. El **Timeout** después del cual el reequilibrio puede interrumpir la actividad de la aplicación
3. **BalanceOptions** específicas

Ejemplos de cuándo se puede volver a equilibrar la aplicación

Ejemplo 1

Ha escrito una aplicación que coloca mensajes bajo un punto de sincronización y confirma el lote de mensajes emitiendo una llamada MQCMIT. Cuando se completa la llamada MQCMIT, la aplicación empieza a colocar mensajes bajo un nuevo punto de sincronización.

Configuración de IBM MQ sugerida
Opciones predeterminadas suficientes

Resultado
Una instancia de aplicación se mueve después de que una llamada MQCMIT tenga éxito (o falle), una vez que se haya cumplido el número configurado de transacciones.

De forma predeterminada, si un lote de mensajes supera los 10 segundos, puede retrotraerse si se ha solicitado un reequilibrio. Si espera que las transacciones superen regularmente este límite y requieren que se permita, puede ampliar **Timeout** de forma adecuada.

Ejemplo 2

Ha escrito una aplicación que coloca un mensaje en una instancia de cola de clúster y otra aplicación responde a una cola dinámica temporal local con un mensaje, después de procesar la solicitud. Cuando la solicitud se ha leído de forma destructiva desde la cola local, la aplicación coloca su siguiente mensaje de solicitud.

Configuración de IBM MQ sugerida
Establecer Type en MQBNO_BALTYPE_REQREP

Resultado
Una instancia de aplicación se mueve cuando una aplicación completa una llamada MQGET, momento en el que la instancia de la aplicación se mueve a otro gestor de colas. Las llamadas MQPUT posteriores se llevan a cabo en el nuevo gestor de colas.

MQBNO
ApplicationType

Multi *Influir en el reequilibrio de aplicaciones en .NET*
En IBM MQ 9.3.0, hay disponibles constantes adicionales para establecer las propiedades de la opción de equilibrio utilizando una tabla hash de la aplicación cuando utilice la clase `MQQueueManager` para conectarse al gestor de colas.

Las constantes siguientes son las que utiliza para influir en el equilibrio de aplicaciones en .NET:

Volver a equilibrar el tipo de aplicación
El tipo de acción de equilibrio; representado por la constante **MQC.BALANCING_APPLICATION_TYPE_PROPERTY**

- Debe utilizar esta propiedad para establecer el campo **ApplicationType** de la estructura MQBNO.

  Debe establecer valores de tipo entero y los valores posibles son:

  MQC.BALANCING_APPLICATION_TYPE_SIMPLE
  Equilibrio simple; no se aplican reglas específicas además de las descritas en "Influir en el reequilibrio de aplicaciones en clústeres uniformes" en la página 439. Éste es el valor predeterminado.

MQC.BALANCING_APPLICATION_TYPE_REQUEST_REPLY
Equilibrio de solicitud-respuesta; después de cada llamada de **MQPUT**, se espera una llamada **MQGET** coincidente para un mensaje de respuesta. El equilibrio se retrasa hasta que se recibe un mensaje de este tipo o se ha excedido el mensaje de solicitud **EXPIRY** .

Si la reconexión está habilitada por la aplicación y esta propiedad no está establecida, se utiliza **MQC.BALANCING_APPLICATION_TYPE_SIMPLE**

Opciones de reequilibrio
Las opciones de equilibrio establecidas por la aplicación emisora; representadas por la constante **MQC.BALANCING_OPTIONS_PROPERTY**

- Debe utilizar esta propiedad para establecer el campo **BalanceOptions** de la estructura MQBNO.

  Debe establecer valores de tipo entero y los valores posibles son:

  MQC.BALANCING_OPTIONS_NONE
  No se han establecido opciones. Se trata del valor predeterminado

  MQC.BALANCING_OPTIONS_IGNORE_TRANSACTIONS
  Establecer esta opción permite que las aplicaciones se reequilibren incluso si están en medio de una transacción.

  Si la reconexión está habilitada por la aplicación y esta propiedad no está establecida, se utiliza **MQC.BALANCING_OPTIONS_NONE**.

Tiempo de espera de reequilibrio
Tiempo de espera tras el cual el reequilibrio podría interrumpir la actividad de la aplicación; representado por la constante **MQC.BALANCING_TIMEOUT_PROPERTY**

- Debe utilizar esta propiedad para establecer el campo Timeout de la estructura MQBNO.

  Debe establecer valores de tipo entero y los valores posibles son:

  MQC.BALANCING_TIMEOUT_AS_DEFAULT
  El valor de tiempo de espera predeterminado establecido. Se trata del valor predeterminado

  MQC.BALANCING_TIMEOUT_IMMEDIATE
  Se produce un tiempo de espera inmediato

  MQC.BALANCING_TIMEOUT_NEVER
  No se produce ningún tiempo de espera

  Nota: Debe proporcionar un valor sólo a partir de los valores definidos o un valor de 0-999999999 segundos.

Despliegue de aplicaciones de cliente flexibles y escalables
MQBNO

Multi V 9.4.0 *Influir en el reequilibrio de aplicaciones en IBM MQ classes for JMS*
A partir de IBM MQ 9.4.0, hay constantes adicionales disponibles para que pueda establecer las propiedades de la opción de equilibrio en un **ConnectionFactory**. Estas constantes sólo son aplicables si **WMQ_PROVIDER_VERSION** se establece en 7. Las aplicaciones de `Request_reply` en un clúster uniforme deben permitir la posibilidad de respuestas perdidas.

- “Las constantes disponibles” en la página 443.
- “El potencial de pérdida de mensajes en el equilibrio de aplicaciones de REQUEST_REPLY” en la página 445.

Las constantes disponibles

Las constantes siguientes son las que utiliza para influir en el equilibrio de aplicaciones en IBM MQ classes for JMS:

Volver a equilibrar el tipo de aplicación
El tipo de acción de equilibrio; representado por la constante **WMQConstants.WMQ_BALANCING_APPLICATION_TYPE**

- Debe utilizar esta propiedad para establecer el campo **ApplicationType** de la estructura MQBNO .

  Debe establecer valores de tipo entero. Estos son los valores posibles:

  WMQConstants.WMQ_BALANCING_APPLICATION_TYPE_SIMPLE (Valor predeterminado)
  Equilibrio simple; no se aplican reglas específicas además de las descritas en "Influir en el reequilibrio de aplicaciones en clústeres uniformes" en la página 439.

  WMQConstants.WMQ_BALANCING_APPLICATION_TYPE_REQUEST_REPLY
  Equilibrio de solicitud-respuesta; después de cada llamada de **MQPUT**, se espera una llamada **MQGET** coincidente para un mensaje de respuesta. El equilibrio se retrasa hasta que se recibe un mensaje de este tipo o se ha excedido el mensaje de solicitud **EXPIRY** .

  Si la aplicación habilita la reconexión y no se establece esta propiedad, se utiliza **WMQConstants.WMQ_BALANCING_APPLICATION_TYPE_SIMPLE** .

Opciones de reequilibrio
Las opciones de equilibrio establecidas por la aplicación emisora; representadas por la constante **WMQConstants.WMQ_BALANCING_OPTIONS**

- Debe utilizar esta propiedad para establecer el campo **BalanceOptions** de la estructura MQBNO .

  Debe establecer valores de tipo entero. Estos son los valores posibles:

  WMQConstants.WMQ_BALANCING_OPTIONS_NONE (Valor predeterminado)
  No se han establecido opciones.

  WMQConstants.WMQ_BALANCING_OPTIONS_IGNORE_TRANSACTIONS
  Establecer esta opción permite que las aplicaciones se reequilibren incluso si están en medio de una transacción.

  Si la aplicación habilita la reconexión y no se establece esta propiedad, se utiliza **WMQConstants.WMQ_BALANCING_OPTIONS_NONE** .

Tiempo de espera de reequilibrio
El tiempo de espera después del cual el reequilibrio puede interrumpir la actividad de la aplicación; representado por la constante **WMQConstants.WMQ_BALANCING_TIMEOUT**

- Debe utilizar esta propiedad para establecer el campo **Timeout** de la estructura MQBNO .

  Debe establecer valores de tipo entero. Estos son los valores posibles:

  WMQConstants.WMQ_BALANCING_TIMEOUT_AS_DEFAULT (Valor predeterminado)
  El valor de tiempo de espera predeterminado establecido. De forma predeterminada, este valor es de 10 segundos.

  WMQConstants.WMQ_BALANCING_TIMEOUT_IMMEDIATE
  Se produce un tiempo de espera inmediato.

  WMQConstants.WMQ_BALANCING_TIMEOUT_NEVER
  No se produce ningún tiempo de espera.

  un valor entre 1 y 999999999
  Representa un valor en segundos.

  Nota: Debe proporcionar un valor sólo a partir de los valores definidos o un valor de 0-999999999 segundos.

Estas propiedades también se pueden establecer en las representaciones JNDI de las fábricas de conexiones utilizando las interfaces JMSAdmin o IBM MQ Explorer .

El potencial de pérdida de mensajes en el equilibrio de aplicaciones de REQUEST_REPLY

En IBM MQ classes for JMS (y IBM MQ classes for Jakarta Messaging), la funcionalidad de solicitud/respuesta se implementa estableciendo la propiedad **JMSReplyTo** en el mensaje de solicitud, que utiliza la aplicación de respuesta para determinar si se envía la respuesta. En términos de JMS , la propiedad **JMSReplyTo** es un **Destination**.

Cuando esto se convierte en operaciones IBM MQ , la propiedad **JMSReplyTo** se envía como un URI de cola completo, que identifica una cola en un gestor de colas específico.

Debido a la naturaleza asíncrona del manejo de reconexiones de equilibrio, es posible que se inicie una reconexión después de que la propiedad **JMSReplyTo** se haya convertido en un URI completo, pero antes de que el mensaje de solicitud se haya colocado en la cola de solicitudes. En estas circunstancias, la aplicación que responde puede enviar su respuesta a la cola de respuestas original en el gestor de colas original, pero la aplicación solicitante podría estar ahora a la espera de una respuesta en el nuevo gestor de colas.

Por lo tanto, las aplicaciones `Request_reply` en un clúster uniforme deben permitir la posibilidad de respuestas perdidas.

Despliegue de aplicaciones de cliente flexibles y escalables
MQBNO-Opciones de equilibrio

Multi *Limitaciones y consideraciones para los clústeres uniformes*

Limitaciones y otros puntos a tener en cuenta al configurar los clústeres uniformes.

Nota: Para obtener los requisitos generales al configurar clústeres uniformes, consulte también “Creación de un nuevo clúster uniforme” en la página 448.

Importancia de la uniformidad entre gestores de colas

De forma predeterminada, cualquier aplicación que se declare como `reconnectable` puede reequilibrarse en un gestor de colas alternativo en un clúster uniforme en cualquier momento. Esto significa que cualquier recurso, por ejemplo, cola, tema o registro de autorización que requieran dichas aplicaciones debe declararse en todos los gestores de colas del clúster uniforme.

La coherencia de la configuración del gestor de colas no está vigilada. Es el administrador del sistema el que debe configurar los miembros del clúster de modo que tengan una configuración similar.

Sin embargo, puede ayudar a la coherencia utilizando la prestación Configuración automática de un script MQSC durante el inicio para compartir scripts MQSC que definen objetos para el clúster y, por lo tanto, asegurarse de que todos tienen las mismas definiciones. Para obtener más información, consulte “Creación de un nuevo clúster uniforme” en la página 448.

Esta uniformidad se extiende a los gestores de colas de repositorio completo para el clúster. Aunque para los clústeres de IBM MQ tradicionales, a menudo se considera una práctica recomendada separar los repositorios completos en sistemas autónomos, en un clúster uniforme, el modelo es que los repositorios completos participan completamente en las cargas de trabajo de clúster y de aplicación de proceso junto con otros nodos.

Solapamiento de clústeres uniformes y clústeres de IBM MQ tradicionales

Un gestor de colas de clúster uniforme puede participar como máximo en un clúster uniforme, y también puede ser miembro de cualquier número de clústeres estándar de IBM MQ. Puede ser útil pensar que el clúster uniforme actúa como un único gestor de colas en el clúster más amplio.

Tenga en cuenta las consideraciones siguientes:

- Un gestor de colas de clúster uniforme que actúe como repositorio completo solo debe ser un repositorio completo para el propio clúster uniforme.

- Del mismo modo, los gestores de colas de repositorio parcial que son miembros de un clúster uniforme, pero que también pueden pertenecer a un clúster IBM MQ tradicional más amplio, no se pueden utilizar como repositorio fuera del clúster uniforme.

  Para obtener más información, consulte Cómo elegir gestores de colas de clúster para contener repositorios completos.

La razón es que los gestores de colas que son repositorios completos para una combinación de clústeres IBM MQ tradicionales y clústeres uniformes, fomentan una divergencia de datos contenidos en la memoria caché de clúster entre los miembros del clúster uniforme y, por lo tanto, se mueven en contra del uso de la característica de clúster uniforme según lo previsto.

Para sustituir un único gestor de colas de repositorio completo por un clúster uniforme, separe el repositorio completo del trabajo de aplicación que está en curso en él y mueva sólo el trabajo de aplicación al clúster uniforme.

Cuando utiliza definiciones automáticas para clústeres uniformes, los canales de clúster no se pueden compartir para su uso en otros clústeres, es decir, establece el atributo **CLUSTER** en el clúster automático y el atributo **CLUSNL** debe estar vacío.

Consideraciones sobre el equilibrio de aplicaciones

Las instancias de aplicación no siempre están equilibradas de forma uniforme, especialmente en las siguientes circunstancias:

- Cuando hay menos instancias de aplicación que gestores de colas en el clúster.
- Durante un breve periodo de tiempo después de que las aplicaciones cliente se conecten o salgan del clúster.

Para evitar que las aplicaciones cliente se reequilibren con demasiada frecuencia, especialmente cuando las conexiones de las aplicaciones van y vienen, se establecen límites sobre la frecuencia con la que el clúster uniforme solicita que se reequilibren las aplicaciones cliente. Después de un periodo de alta actividad de conexión o desconexión, las instancias de aplicación restantes pueden tardar varios minutos en equilibrarse uniformemente en el clúster uniforme.

Para obtener más información, consulte Resolución de problemas de equilibrio de aplicaciones.

Afinidades de aplicación

No todas las aplicaciones son adecuadas para el reequilibrio automático en un clúster uniforme. Sólo se reequilibran las aplicaciones que especifican **MQCNO_RECONNECT** . Las aplicaciones que tienen una afinidad con un gestor de colas determinado deben especificar la opción **MQCNO_NO_RECONNECT** o **MQCNO_RECONNECT_Q_MGR**. Este último permite la migración tras error de HA pero no el reequilibrio.

Ejemplos de aplicaciones que crean una afinidad implícita con un gestor de colas:

- Aplicaciones que crean suscripciones duraderas.
- Aplicaciones que crean colas dinámicas permanentes, por ejemplo, para recibir mensajes de respuesta.
- Las aplicaciones que esperan un orden de mensajes estricto o que requieren que todos los mensajes de una secuencia sean procesados por la misma instancia de aplicación o por ambas.

Estas aplicaciones deben especificar opciones **MQCNO_NO_RECONNECT** o **MQCNO_RECONNECT_Q_MGR** en lugar de **MQCNO_RECONNECT**.

Para obtener más información, consulte Opciones de reconexión.

Disponibilidad de mensajes

Aunque el equilibrio de aplicaciones puede reequilibrar las conexiones en torno a gestores de colas anómalos o temporalmente no disponibles, los clústeres uniformes no replican los datos de mensajes entre sus miembros. Para la disponibilidad de datos, si un nodo falla, cada miembro del clúster uniforme también se debe configurar para que sea de alta disponibilidad. Hay muchas soluciones de réplica de

datos y alta disponibilidad disponibles, y se pueden combinar con clústeres uniformes para obtener la máxima disponibilidad de servicios y datos, por ejemplo:

- Almacenamiento replicado que da soporte a una instancia de contenedor que se reinicia automáticamente mediante la orquestación de contenedor. Para obtener más información, consulte Gestor de colas resiliente único.
- Gestores de colas RDQM. Para obtener más información, consulte Alta disponibilidad de RDQM.
- Gestores de colas de varias instancias. Para obtener más información, consulte “Gestores de colas multiinstancia” en la página 530.
- HA nativa. Para obtener más información, consulte HA nativa.
- IBM MQ Appliance HA. Para obtener más información, consulte Alta disponibilidad.

Escalabilidad y rendimiento de clústeres uniformes

Para permitir una integración y una compartición más estrechas del estado de la aplicación entre gestores de colas de un clúster uniforme, se necesita un nivel más alto de intercomunicación que en un clúster IBM MQ tradicional. Por lo tanto, no se recomienda escalar a un gran número de gestores de colas en un único clúster uniforme porque la comunicación adicional tiene un efecto perjudicial en el rendimiento.

Por razones de rendimiento y gestión, es preferible pensar que un clúster uniforme actúa como un único gestor de colas tradicional que proporciona mensajería a una serie de aplicaciones relacionadas, pero no es un único servicio de mensajería en una empresa. En este patrón, los pequeños números, hasta 10, los gestores de colas suelen ser suficientes para dar soporte a un gran número de conexiones de aplicaciones cliente. El equilibrio de aplicaciones hace que sea sencillo empezar con números pequeños, por ejemplo 3 gestores de colas, y escalar añadiendo más gestores de colas.

Atención: la habilitación de un comportamiento de clúster uniforme en un clúster que no tenga las características recomendadas, en particular, el uso de clústeres con un gran número de gestores de colas, es probable que tenga un impacto grave en el rendimiento.

Conceptos relacionados

“Equilibrio de aplicaciones automático” en la página 434
El equilibrado automático de aplicaciones mejora considerablemente la distribución y la disponibilidad de las aplicaciones habilitando un clúster uniforme de IBM MQ para gestionar de cerca la distribución de aplicaciones en todo el clúster, en lugar de depender de la aleatorización o de un anclaje manual de aplicaciones a gestores de colas específicos.

ALW Creación de un clúster uniforme

Puede simplificar la creación inicial de un clúster uniforme y, posteriormente, hacer que la configuración de los miembros del clúster uniforme sea idéntica, utilizando el soporte de configuración automática y agrupación en clúster automática.

Antes de empezar

Antes de crear un clúster uniforme, debe leer “Limitaciones y consideraciones para los clústeres uniformes” en la página 445.

Acerca de esta tarea

Indica que un clúster de IBM MQ determinado se debe tratar como un clúster uniforme proporcionando en el archivo `qm.ini` una sección para AutoCluster con al menos **Type**=*Uniform* y **ClusterName**=*< nombre de clúster uniforme >*.

Opcionalmente, puede configurar el clúster IBM MQ subyacente a través de la misma stanza `.ini` utilizando *automatic cluster creation*. Cuando se utiliza este soporte de clúster automático para configurar el clúster, se proporciona un archivo de configuración que describe el clúster y sus repositorios completos.

Si el gestor de colas que se está iniciando aparece como uno de los repositorios completos, se crea automáticamente un repositorio completo. De forma similar, cuando se define el canal receptor de clúster, los canales emisores de clúster al repositorio o repositorios completos se definen automáticamente.

Procedimiento

Para utilizar una función adicional que requiere un clúster uniforme, debe completar uno de los pasos siguientes:

- Convierta un clúster existente en un clúster uniforme, que cumple el patrón descrito en “Acerca de los clústeres uniformes” en la página 432.
- Cree un nuevo clúster uniforme para esta finalidad.

ALW *Creación de un nuevo clúster uniforme*

Cómo crear un nuevo clúster uniforme.

Procedimiento

1. Cree un archivo que describa cómo desea que sea el propio clúster en términos de repositorios completos.

   Como cualquier clúster, dos repositorios completos actúan como almacenes centrales de información sobre el clúster.

   Específicamente, debe describir los nombres y los nombres de conexión para los dos repositorios completos en este clúster.

   Nota: Esto se realiza antes de la creación de cualquier otro elemento (incluidos los gestores de colas) y el proceso siguiente, que se muestra a continuación, incluye la creación de los gestores de colas.

   Por ejemplo, imagine que está configurando un clúster uniforme denominado UNICLUS, con los miembros del gestor de colas QMA, QMB, QMC y QMD. En este ejemplo, QMA y QMB serán los repositorios completos, con QMC y QMD como repositorios parciales. Un archivo de configuración de ejemplo, `uniclus.ini`:

   ```
   AutoCluster:
      Repository2Conname=QMA.dnsname(1414)
      Repository2Name=QMA
      Repository1Conname=QMB.dnsname(1414)
      Repository1Name=QMB
      ClusterName=UNICLUS
      Type=Uniform
   ```

   Los campos **RepositoryNConname** se utilizan como el atributo *conname* para que otros miembros del clúster definan los remitentes de clúster (CLUSSDR) para ellos, y puede ser una lista de conexiones para un gestor de colas de varias instancias y opcionalmente puede incluir el puerto.
2. Cree un archivo de configuración de ejemplo, `uniclus.mqsc` que contenga las definiciones MQSC que desea aplicar a todos los miembros del clúster.

   Hay una línea obligatoria necesaria en este archivo, que es una definición de un canal receptor de clúster (CLUSRCVR), con un atributo CLUSTER del nombre de clúster automático (normalmente mediante la inserción de +AUTOCL+) y un nombre de canal que incluye la inserción +QMNAME+.

   Esto describe cómo otros miembros del clúster uniforme se conectan a cada gestor de colas y se utilizan como una plantilla de cómo conectarse también a los otros gestores de colas. Una definición de ejemplo puede ser algo parecido a:

   ```
   define channel('+AUTOCL+_+QMNAME+') chltype(clusrcvr) trptype(tcp)
   conname(+CONNAME+) cluster('+AUTOCL+') replace
   ```

Cuando se configuran clústeres automáticos, una definición de un canal receptor de clúster puede utilizar algunas inserciones adicionales en los campos CLUSTER, CONNAME y CHANNEL para permitir que la definición sea idéntica en todos los gestores de colas del clúster uniforme. Esto incluye:

+AUTOCL+
Nombre automático del clúster

+QMNAME+
Nombre del gestor de colas que se está creando

+CONNAME+
Variable definida durante la creación del gestor de colas, utilizando el parámetro **-iv** o en la stanza `Variables` qm.ini, para su uso en la serie de parámetro de nombre de conexión. El nombre de la variable puede ser cualquier valor.

Recuerde que los nombres de canal tienen una limitación de 20 caracteres, por lo que tanto el valor con las inserciones como cuando se sustituyen las inserciones, debe ajustarse a dicha limitación. Un archivo de ejemplo puede tener el aspecto siguiente:

```
*##########################################################################
* Compulsory section for all uniform cluster queue managers
*##########################################################################
define channel('+AUTOCL+_+QMNAME+') chltype(clusrcvr) trptype(tcp) conname(+CONNAME+)
cluster('+AUTOCL+') replace
*
*##########################################################################
* Configuration for all queue managers
*##########################################################################
define QL(APPQ) maxdepth(99999999) replace
define QL(APPQ2) maxdepth(99999999) replace
define channel(CLIENTCHL) chltype(svrconn) trptype(tcp) replace
```

3. Haga que estos dos archivos estén disponibles en cada una de las máquinas que alojarán un miembro de clúster uniforme.

   Por ejemplo, `/shared/uniclus.ini` y `/shared/uniclus.mqsc`.

4. En cada una de estas máquinas, cree el gestor de colas.

   En la línea de mandatos, proporcione:

   a. Una solicitud para iniciar un escucha, en el puerto esperado

   b. Una solicitud de configuración INI automática (**-ii**) que apunta al archivo de configuración de clúster automático (uniclus.ini)

   c. Una solicitud de configuración MQSC automática (**-ic**) que apunta al archivo de configuración MQSC que incluye una definición de CLUSRCVR para el clúster uniforme.

   d. Un CONNAME para este gestor de colas.

   En el host para QMA:

```
crtmqm -p 1414 -ii /shared/uniclus.ini -ic /shared/uniclus.mqsc -iv
CONNAME=QMA.dnsname(1414) QMA
strmqm QMA
```

   Cada gestor de colas del clúster uniforme se crea con una línea de mandatos casi idéntica: todas las diferencias entre el repositorio completo y parcial se gestionan automáticamente para un clúster uniforme.

   En el host para QMB:

```
crtmqm -p 1414 -ii /shared/uniclus.ini -ic /shared/uniclus.mqsc -iv
CONNAME=QMB.dnsname(1414) QMB
strmqm QMB
```

   En el host para QMC:

```
crtmqm -p 1414 -ii /shared/uniclus.ini -ic /shared/uniclus.mqsc -iv
CONNAME=QMC.dnsname(1414) QMC
strmqm QMC
```

En el host para QMD:

```
crtmqm -p 1414 -ii /shared/uniclus.ini -ic /shared/uniclus.mqsc -iv
CONNAME=QMD.dnsname(1414) QMD
strmqm QMD
```

Qué sucede automáticamente:

A medida que se inicia el gestor de colas, las definiciones del archivo `uniclus.ini` se aplican al archivo `qm.ini`. Para obtener más información, consulte “Configuración automática de qm.ini en el inicio” en la página 110. Esto añade la definición **AutoCluster** al archivo `qm.ini`.

Si el gestor de colas se denomina en la stanza **AutoCluster** como uno de los repositorios completos, se convierte automáticamente para que sea un repositorio completo, similar a emitir el mandato MQSC ALTER QMGR REPOS (**ClusterName**), de lo contrario se convierte en un repositorio parcial, similar a emitir el mandato MQSC ALTER QMGR REPOS ('').

Cuando se procesa la definición del canal receptor del clúster para el clúster automático, los canales de emisor del clúster se definen desde este gestor de colas a todos los repositorios completos de la stanza **AutoCluster** (excluyendo el gestor de colas local si este es uno de los repositorios completos). Estos canales emisores heredan todos los atributos de canal comunes del receptor de clúster local que se ha definido.

Atención: Aunque los canales se crean sin una intervención manual adicional, estos son objetos de canal administrativo que se pueden visualizar y gestionar como para cualquier otra definición de canal. No debe confundir estos objetos con canales emisores de clúster 'autodefinidos', creados transitoriamente y bajo demanda por el clúster para direccionar el tráfico de mensajes.

Qué hacer a continuación

Verificar la configuración de clúster uniforme

Cuando el parámetro **ClusterName** se establece correctamente, y el gestor de colas es miembro del clúster con nombre, se emite el mensaje AMQ9883 para confirmar que el clúster se identifica ahora como un clúster uniforme.

A continuación, puede utilizar funciones de clúster uniforme como, por ejemplo, el equilibrado de aplicaciones automático. Durante el inicio del gestor de colas, si se ha establecido este parámetro, pero el nombre no es un nombre de clúster de IBM MQ válido, se omite el nombre y se emite el mensaje de error AMQ9882.

Si el nombre es un nombre de clúster válido, pero no existen canales de clúster para el clúster identificado, se emite el mensaje de aviso AMQ9881 al registro de errores del gestor de colas para permitir que el administrador identifique y corrija esta situación.

Verificar la configuración de clúster automatizado

Si ha utilizado el soporte de clúster automático para configurar el clúster uniforme, puede verificar que los gestores de colas especificados como repositorios completos ahora están configurados correctamente como tales, utilizando mandatos runmqsc:

```
QMA:
     1 : dis qmgr repos
AMQ8408I: Display Queue Manager details.
   QMNAME(QMA)                             REPOS(UNICLUS)
```

Mientras que los repositorios parciales no están configurados como repositorios:

```
QMC:
     1 : dis qmgr repos
AMQ8408I: Display Queue Manager details.
   QMNAME(QMC)                             REPOS( )
```

Además, debe poder ver que los canales emisores de clúster (CLUSSDR) se han configurado desde cada gestor de colas a los demás repositorios completos, utilizando el nombre de canal del archivo de configuración MQSC:

```
QMA:
     1 : dis chl(UNICLUS*) conname
AMQ8414I: Display Channel details.
   CHANNEL(UNICLUS_QMA)                    CHLTYPE(CLUSRCVR)
   CONNAME(QMA.dnsname(1414))
AMQ8414I: Display Channel details.
   CHANNEL(UNICLUS_QMB)                    CHLTYPE(CLUSSDR)
   CONNAME(QMB.dnsname(1414))

QMC:
     1 : dis chl(UNICLUS*) conname
AMQ8414I: Display Channel details.
   CHANNEL(UNICLUS_QMA)                    CHLTYPE(CLUSSDR)
   CONNAME(QMA.dnsname(1414))
AMQ8414I: Display Channel details.
   CHANNEL(UNICLUS_QMB)                    CHLTYPE(CLUSSDR)
   CONNAME(QMB.dnsname(1414))
AMQ8414I: Display Channel details.
   CHANNEL(UNICLUS_QMC)                    CHLTYPE(CLUSRCVR)
   CONNAME(QMC.dnsname(1414))
```

Conceptos relacionados

“Acerca de los clústeres uniformes” en la página 432
El objetivo de un despliegue de clúster uniforme es que las aplicaciones se puedan diseñar para la escala y la disponibilidad, y que puedan conectarse a cualquiera de los gestores de colas dentro del clúster uniforme. Esto elimina cualquier dependencia de un gestor de colas específico, lo que produce una mejor disponibilidad y equilibrio de carga de trabajo del tráfico de mensajería. z/OS Los clústeres uniformes no están disponibles en IBM MQ for z/OS, los grupos de compartición de colas proporcionan muchas de las prestaciones de un clúster uniforme.

“Limitaciones y consideraciones para los clústeres uniformes” en la página 445
Limitaciones y otros puntos a tener en cuenta al configurar los clústeres uniformes.

ALW *Convertir un clúster existente en un clúster uniforme*

Puede utilizar este procedimiento para convertir un clúster existente en un clúster uniforme.

Acerca de esta tarea

Si convierte un clúster existente en un clúster uniforme, debe asegurarse de que exista cualquier definición necesaria para dar soporte al equilibrio de aplicaciones entre los gestores de colas en todos los miembros del clúster.

Procedimiento

1. Habilitar la suscripción de publicación de IBM MQ, incluida la suscripción de publicación remota (en clúster) en todos los gestores de colas.

   Este es un requisito previo para la funcionalidad de clúster uniforme, por lo tanto, debe asegurarse de que los atributos PSMODE y PSCLUS del gestor de colas estén establecidos en el valor predeterminado de ENABLED.
2. Añada una sección **AutoCluster** en el archivo `qm.ini` en el nombre del clúster de IBM MQ, tal como se utiliza en las definiciones de objeto de MQSC, como por ejemplo, los canales de clúster.

Por ejemplo, si el nombre del clúster es UNICLUS, añada o modifique la stanza AutoCluster en los archivos `qm.ini` tal como se indica a continuación:

```
AutoCluster:
      ClusterName=UNICLUS
    Type=Uniform
```

3. Reinicie los gestores de colas para aplicar el nuevo valor.
4. Considere la posibilidad de utilizar la configuración automática como mecanismo para garantizar que se aplique desde el inicio la misma configuración a todos los miembros de clúster uniforme.

   Para obtener más información, consulte Configuración automática desde un script MQSC durante el inicio.

Conceptos relacionados

“Acerca de los clústeres uniformes” en la página 432
El objetivo de un despliegue de clúster uniforme es que las aplicaciones se puedan diseñar para la escala y la disponibilidad, y que puedan conectarse a cualquiera de los gestores de colas dentro del clúster uniforme. Esto elimina cualquier dependencia de un gestor de colas específico, lo que produce una mejor disponibilidad y equilibrio de carga de trabajo del tráfico de mensajería. z/OS Los clústeres uniformes no están disponibles en IBM MQ for z/OS, los grupos de compartición de colas proporcionan muchas de las prestaciones de un clúster uniforme.

“Limitaciones y consideraciones para los clústeres uniformes” en la página 445
Limitaciones y otros puntos a tener en cuenta al configurar los clústeres uniformes.

Multi Utilización de la configuración de clúster automático

Puede configurar IBM MQ para habilitar la configuración automática cambiando la información de configuración de `qm.ini`.

Nota: Solo puede utilizar la stanza AutoCluster para clústeres uniformes.

Stanzas para configurar

Puede cambiar las stanzas siguientes:

AutoConfig
Definido en el archivo `qm.ini`. Cuando se inicia el gestor de colas, identifica los archivos de configuración automática que se deben aplicar.

Debe utilizar este mecanismo para distribuir una configuración de clúster idéntica cuando utilizan clústeres uniformes.

AutoCluster
Definido en el archivo `qm.ini`. Se utiliza cuando se inicia el gestor de colas para identificar si el clúster es miembro de un clúster automático y puede identificar los repositorios completos del clúster.

Variables
Definido en el archivo `qm.ini`. Contiene algunas variables del gestor de colas.

Atributos de la stanza AutoConfig

Se permiten los dos atributos siguientes en la stanza AutoConfig:

MQSCConfig=*<Path>*
La vía de acceso es una vía de acceso de archivo completa o una vía de acceso a un directorio, donde todos los archivos `*.mqsc` se aplican al gestor de colas, en cada inicio del gestor de colas.

Para obtener más información, consulte Configuración automática de un script de mandato de script de WebSphere MQ en el inicio.

IniConfig=*<Path>*
La vía de acceso es una vía de acceso de archivo completa o una vía de acceso a un directorio, donde todos los archivos `*.ini` se aplican al archivo `qm.ini`, en cada inicio del gestor de colas.

Para obtener más información, consulte "Configuración automática de qm.ini en el inicio" en la página 110.

Estos atributos se utilizan con frecuencia como parte de la configuración de clústeres uniformes. Para obtener más información, consulte "Creación de un nuevo clúster uniforme" en la página 448.

Stanza de ejemplo:

```
AutoConfig:
MQSCConfig=C:\MQ_Configuration\uniclus.mqsc
IniConfig=C:\MQ_Configuration\uniclus.ini
```

Atributos de la stanza AutoCluster

Los atributos siguientes son obligatorios para la stanza AutoCluster:

Type=*Uniform*
Especifica el tipo de clúster automático y la única opción válida es *Uniform*, que representa un clúster uniforme.

ClusterName=*<String>*
Nombre del clúster, que es el nombre de clúster automático.

La presencia de los atributos anteriores permite el equilibrio de aplicaciones para los clústeres uniformes. Para obtener información más detallada, consulte el apartado "Equilibrio de aplicaciones automático" en la página 434.

Además, se puede realizar la configuración simplificada de un clúster si se describe el clúster en esta stanza. Para obtener más información, consulte "Creación de un nuevo clúster uniforme" en la página 448. Cuando se utiliza esto, puede dar nombre a dos gestores de colas y dar sus nombres de conexión para los repositorios completos para este clúster automático.

Los atributos siguientes son opcionales para la stanza AutoCluster, pero debe proporcionarlos en pares:

RepositoryName1 =*< Serie>*
Se trata del nombre de gestor de colas para el primer repositorio completo del clúster automático. Este puede ser el nombre de este gestor de colas o de otro.

Repository1Conname=*< Serie de nombre de conexión >*
Se trata del valor de nombre de conexión (CONNAME) para la forma en que los miembros del clúster automático deben conectarse a este gestor de colas.

Además, puede identificar un segundo repositorio completo para el clúster:

Repository2Name=*< Serie>*
Repository2Conname=*< Serie de nombre de conexión >*

Stanza de ejemplo:

```
AutoCluster:
   Repository2Conname=myFR1.hostname(1414)
   Repository2Name=QMFR1
   Repository1Conname= myFR2.hostname(1414)
   Repository1Name=QMFR2
   ClusterName=UNICLUS
   Type=Uniform
```

Atributos de la stanza Variables

Un par `attribute=value` es válido en el campo de atributo. Se pueden proporcionar utilizando la opción de línea de mandatos **-iv** en el mandato **crtmqm** al crear un gestor de colas.

Puede utilizar los atributos enumerados en la stanza Variables durante la configuración automática de clúster de CONNAME y los campos de mandato de script de IBM MQ de nombre de canal de un canal de clúster receptor.

Suspensión de un gestor de colas de un clúster uniforme

Durante el funcionamiento normal de un clúster uniforme, las instancias de aplicación cliente reconectables se pueden reequilibrar automáticamente en cualquier momento, en cualquier gestor de colas del clúster. Si desea impedir que las aplicaciones se conecten a un gestor de colas determinado durante un período de tiempo, por ejemplo, durante las operaciones de mantenimiento o la determinación de problemas, utilice el mandato SUSPEND QMGR.

Emita el mandato SUSPEND QMGR CLUSTER(*nombre de clúster uniforme*)

Además de los efectos habituales de la suspensión desde un clúster de IBM MQ , en un clúster uniforme, el mandato SUSPEND también impide que las aplicaciones reconectables se reequilibren a este gestor de colas.

Las conexiones existentes de este tipo con el gestor de colas se vuelve a equilibrar inmediatamente con otros gestores de colas disponibles en el clúster cuando se emite el mandato.

Notas:

- Cuando los gestores de colas se suspenden de un clúster, DIS APSTATUS los muestra como ACTIVE (NO), con la excepción del gestor de colas local, que siempre muestra ACTIVE (YES) para su propia entrada de estado.
- si todos los gestores de colas del clúster uniforme están suspendidos, las aplicaciones permanecen conectadas a uno o varios de los gestores de colas suspendidos.

Para evitar que se añadan nuevas conexiones al gestor de colas que se mantiene, debe detener el canal o canales de conexión de servidor utilizados por las aplicaciones cliente, por ejemplo, emitiendo el siguiente mandato **runmqsc** :

```
STOP CHANNEL(svrconn channel name)
```

Esto no sería posible si, por ejemplo, estos canales también se utilizan para conectar aplicaciones administrativas necesarias durante la ventana de mantenimiento. Por este motivo, el gestor de colas suspendido comprueba periódicamente si hay aplicaciones reconectables conectadas

Si hay aplicaciones reconectables, se reequilibran con otros gestores de colas disponibles en el clúster. Ahora se puede realizar el mantenimiento en el gestor de colas suspendido.

Nota: las aplicaciones no consideradas móviles no se ven afectadas ni por el mandato inicial ni por los reescaneos posteriores, y permanecen conectadas al gestor de colas suspendido; consulte MOVCOUNT para obtener más detalles.

Para reanudar un gestor de colas suspendido:

1. Si es necesario, inicie el canal de conexión del servidor para reanudar la aceptación de nuevas conexiones de aplicación, emitiendo el siguiente mandato:

   ```
   START CHANNEL(svrconn channel name)
   ```

2. Emita el siguiente mandato **runmqsc** :

   ```
   RESUME QMGR CLUSTER(uniform cluster name)
   ```

El gestor de colas reanuda la comunicación con el resto del clúster uniforme y, si es necesario para restaurar el equilibrio, las instancias de aplicación cliente reconectables se redirigen a este gestor de colas.

Configurar la mensajería de publicación/suscripción

Puede iniciar, detener y visualizar el estado de la publicación/suscripción en cola. También puede añadir y eliminar corrientes de datos, y añadir y suprimir gestores de colas de una jerarquía de intermediarios.

Procedimiento

- Consulte los subtemas siguientes para obtener más información sobre el control de publicación/suscripción en cola:
  - "Establecimiento de atributos de mensajes de publicación/suscripción en cola" en la página 455
  - "Inicio de la publicación/suscripción en cola" en la página 456
  - "Detención de publicación/suscripción en cola" en la página 457
  - "Adición de una corriente" en la página 457
  - "Supresión de una corriente de datos" en la página 458
  - "Adición de un punto de suscripción" en la página 459
  - "Combinación de espacios de temas en redes de publicación/suscripción" en la página 467

Establecimiento de atributos de mensajes de publicación/suscripción en cola

Puede controlar el comportamiento de algunos atributos de mensajes de publicación/suscripción utilizando atributos del gestor de colas. Los otros atributos que controla en la stanza *Broker* del archivo `qm.ini`.

Acerca de esta tarea

Puede establecer los siguientes atributos de publicación/suscripción; para obtener más detalles, consulte Parámetros del gestor de colas

Tabla 28. Parámetros de configuración de publicación/suscripción

Descripción	**Nombre de parámetro de MQSC**
Cuenta de reintentos de mensaje de mandato	**PSRTYCNT**
Descartar mensaje de entrada de mandato no entregable	**PSNPMSG**
Comportamiento que sigue al mensaje de respuesta de mandato no entregable	**PSNPRES**
Procesar mensajes de mandatos bajo syncpoint	**PSSYNCPT**

La stanza Broker se utiliza para gestionar los siguientes valores de configuración:

- `PersistentPublishRetry=yes | force`

  Si especifica `Yes`, si una publicación de un mensaje persistente a través de la interfaz de publicación/suscripción en cola falla, y no se ha solicitado ninguna respuesta negativa, la operación de publicación se vuelve a intentar.

  Si ha solicitado un mensaje de respuesta negativa, la respuesta negativa se envía y no se produce ningún otro reintento.

  Si especifica `Force`, si una publicación de un mensaje persistente a través de la interfaz de publicación/suscripción falla, la operación de publicación se vuelve a intentar hasta que se procesa satisfactoriamente. No se envía ninguna respuesta negativa.
- `NonPersistentPublishRetry=yes | force`

Si especifica Yes, si una publicación de un mensaje no persistente a través de la interfaz de publicación/suscripción en cola falla y no se solicita ninguna respuesta negativa, la operación de publicación se vuelve a intentar.

Si ha solicitado un mensaje de respuesta negativa, la respuesta negativa se envía y no se produce ningún otro reintento.

Si ha especificado Force, si una publicación de un mensaje no persistente a través de la interfaz de publicación/suscripción en cola falla, la operación de publicación se reintenta hasta que se ha procesado satisfactoriamente. No se envía ninguna respuesta negativa.

Nota: Si desea habilitar esta funcionalidad para mensajes no persistentes, así como establecer el valor NonPersistentPublishRetry también debe asegurarse de que el atributo del gestor de colas **PSSYNCPT** esté establecido en Yes.

Esto también podría tener un impacto en el rendimiento del proceso de publicaciones no persistentes porque **MQGET** de la cola STREAM ahora se produce bajo el punto de sincronismo.

- PublishBatchSize=*número*

  El intermediario normalmente procesa mensajes de publicación dentro del punto de sincronismo. Puede ser ineficaz para confirmar cada publicación de forma individual y, en algunas circunstancia, el intermediario puede procesar varios mensajes de publicación en una sola unidad de trabajo. Este parámetro especifica el número máximo de mensajes de publicación que pueden procesarse en una sola unidad de trabajo.

  El valor predeterminado para PublishBatchSize es 5.

- PublishBatchInterval=*número*

  El intermediario normalmente procesa mensajes de publicación dentro del punto de sincronismo. Puede ser ineficaz para confirmar cada publicación de forma individual y, en algunas circunstancia, el intermediario puede procesar varios mensajes de publicación en una sola unidad de trabajo. Este parámetro especifica el tiempo máximo (en milisegundos) entre el primer mensaje de un lote y cualquier publicación posteriores incluida en el mismo lote.

  Un intervalo de lote 0 indica que se pueden procesar hasta PublishBatchSize mensaje, siempre que los mensajes estén disponibles inmediatamente.

  El valor predeterminado para PublishBatchInterval es cero.

Procedimiento

Utilice IBM MQ Explorer, mandatos programables o el mandato **runmqsc** para modificar los atributos de gestor de colas que controlan el comportamiento de publicación/suscripción.

Ejemplo

ALTER QMGR PSNPRES(SAFE)

Inicio de la publicación/suscripción en cola

Inicie la publicación/suscripción en cola estableciendo el atributo PSMODE del gestor de colas.

Antes de empezar

Lea la descripción de PSMODE para conocer las tres modalidades de publicación/suscripción:

- COMPAT
- Inhabilitado
- Habilitado

Acerca de esta tarea

Establezca el atributo PSMODE de QMGR para iniciar la interfaz de publicación/suscripción en cola (también conocida como el intermediario), o el motor de publicación/suscripción (también conocido como publicación/suscripción de la versión 7) o ambos. Para iniciar la publicación/suscripción en cola necesita establecer PSMODE en ENABLED. El valor predeterminado es ENABLED.

Procedimiento

Utilice IBM MQ Explorer o el mandato **runmqsc** para habilitar la interfaz de publicación/suscripción en cola si la interfaz ya no está habilitada.

Ejemplo

```
ALTER QMGR PSMODE (ENABLED)
```

Qué hacer a continuación

IBM MQ procesa mandatos de publicación/suscripción en cola y llamadas de Interfaz de cola de mensajes (MQI) de publicación/suscripción.

Detención de publicación/suscripción en cola

Puede detener la publicación/suscripción en cola estableciendo el atributo PSMODE del gestor de colas.

Antes de empezar

Lea la descripción de PSMODE para conocer las tres modalidades de publicación/suscripción:

- COMPAT
- DISABLED
- ENABLED

Acerca de esta tarea

Establezca el atributo PSMODE de QMGR para detener la interfaz de publicación/suscripción en cola (también conocida como el intermediario), o el motor de publicación/suscripción (también conocido como publicación/suscripción de la versión 7) o ambos. Para detener publicación/suscripción en cola necesita establecer PSMODE en COMPAT. Para detener completamente el motor de publicación/suscripción, establezca PSMODE en DISABLED.

Procedimiento

Utilice IBM MQ Explorer o el mandato **runmqsc** para inhabilitar la interfaz de publicación/suscripción en cola.

Ejemplo

```
ALTER QMGR PSMODE (COMPAT)
```

Adición de una corriente

Puede añadir corrientes manualmente para permitir el aislamiento de los datos entre aplicaciones o para permitir la interoperación con las jerarquías de publicación/suscripción de la IBM MQ.

Antes de empezar

Familiarícese con la manera en que funcionan las corrientes de publicación/suscripción. Consulte Flujos y temas.

Acerca de esta tarea

Utilice el mandato PCF, **runmqsc**, o IBM MQ Explorer para estos pasos.

Nota: Puede realizar los pasos 1 y 2 en cualquier orden. Realice sólo el paso 3 tras haber finalizado los pasos 1 y 2.

Procedimiento

1. Defina una cola local con el mismo nombre que la corriente en la versión anterior de IBM MQ.
2. Defina un tema local con el mismo nombre que la corriente en la versión ht anterior de IBM MQ.
3. Añada el nombre de la cola a la lista de nombres, SYSTEM.QPUBSUB.QUEUE.NAMELIST
4. Repita esta acción para todos los gestores de colas de la versión posterior de IBM MQ que estén en la jerarquía de publicación/suscripción.

Adición de 'Sport'

En el ejemplo de compartición de la corriente 'Sport', los gestores de colas de la versión anterior y los gestores de colas de la versión posterior IBM MQ están trabajando en la misma jerarquía de publicación/suscripción. Los gestores de colas de la versión anterior comparten una corriente denominada 'Sport'. El ejemplo muestra cómo crear una cola y un tema en gestores de colas de versiones posteriores denominados 'Sport', con una serie de tema 'Sport' que se comparte con la corriente de gestores de colas de versiones anteriores 'Sport'.

Una aplicación de publicación de gestor de colas de versión posterior, que publica en el tema 'Sport', con la serie de tema 'Soccer/Results', crea la serie de tema resultante 'Sport/Soccer/Results'. En los gestores de colas de la versión posterior, los suscriptores del tema 'Sport', con la serie de tema 'Soccer/Results' reciben la publicación.

En gestores de colas de versiones anteriores, los suscriptores de la corriente 'Sport', con la serie de tema 'Soccer/Results' reciben la publicación.

```
runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
define qlocal('Sport')
     1 : define qlocal('Sport')
AMQ8006: IBM MQ queue created.
define topic('Sport') topicstr('Sport')
     2 : define topic('Sport') topicstr('Sport')
AMQ8690: IBM MQ topic created.
alter namelist(SYSTEM.QPUBSUB.QUEUE.NAMELIST) NAMES('Sport', 'SYSTEM.BROKER.DEFAULT.STREAM',
'SYSTEM.BROKER.ADMIN.STREAM')
     3 : alter namelist(SYSTEM.QPUBSUB.QUEUE.NAMELIST) NAMES('Sport', 'SYSTEM.BROKER.DEFAULT.STREAM',
'SYSTEM.BROKER.ADMIN.STREAM')
AMQ8551: IBM MQ namelist changed.
```

Nota: Necesita proporcionar los nombres existentes del objeto namelist, además de los nombres nuevos que está añadiendo al mandato **alter namelist**.

Qué hacer a continuación

La información acerca de la corriente se pasa a otros intermediarios de la jerarquía.

Debe configurar cada gestor de colas de IBM MQ en la jerarquía manualmente.

Supresión de una corriente de datos

Puede suprimir una corriente de un gestor de colas de IBM MQ .

Antes de empezar

Antes de suprimir una corriente debe asegurarse de que no existen suscripciones restantes para la corriente y desactivar temporalmente todas las aplicaciones que utilizan la corriente. Si hay publicaciones

que continúan fluyendo a una corriente suprimida, resulta muy costoso a nivel administrativo restaurar el sistema a un estado totalmente operativo.

Procedimiento

1. Encuentre todos los intermediarios conectados que alojen esta corriente.
2. Cancele todas las suscripciones con la corriente en todos los intermediarios.
3. Elimine la cola (que tenga el mismo nombre que la corriente) de la lista de nombres, `SYSTEM.QPUBSUB.QUEUE.NAMELIST`.
4. Suprima o depure todos los mensajes de la cola que tengan el mismo nombre que la corriente.
5. Suprima la cola que tenga el mismo nombre que la corriente.
6. Suprima el objeto de tema asociado.

Qué hacer a continuación

Repita los pasos del 3 al 5 en todos los demás gestores de colas de IBM MQ conectados que alojen la corriente.

Adición de un punto de suscripción

Amplíe una aplicación de publicación/suscripción en cola existente que haya migrado desde IBM Integration Bus con un nuevo punto de suscripción.

Antes de empezar

1. Compruebe que el punto de suscripción no esté ya definido en `SYSTEM.QPUBSUB.SUBPOINT.NAMELIST`.
2. Compruebe si hay un objeto de tema o una serie de tema con el mismo nombre que el punto de suscripción.

Acerca de esta tarea

IBM MQ, las aplicaciones no utilizan puntos de suscripción, pero pueden interoperar con aplicaciones existentes que sí lo hacen, utilizando el mecanismo de migración de puntos de suscripción.

Importante: El mecanismo de migración de puntos de suscripción se ha eliminado de IBM MQ 8.0. Si necesita migrar a las aplicaciones existente, debe realizar los procedimientos descritos en la documentación para su versión del producto, antes de migrar a la versión anterior.

No es necesario añadir puntos de suscripción para utilizar aplicaciones de publicación/suscripción integradas escritas para versiones de IBM MQ.

Procedimiento

1. Añada el nombre del punto de suscripción a `SYSTEM.QPUBSUB.SUBPOINT.NAMELIST`.
   - z/OS En z/OS, **NLTYPE** es `NONE`, el valor predeterminado.
   - Repita el paso en cada gestor de colas que esté conectado en la misma topología de publicación/suscripción.
2. Añada un objeto de tema, preferiblemente asignándole el nombre del punto de suscripción, con una serie de tema que coincida con el nombre del punto de suscripción.
   - Si el punto de suscripción se encuentra en un clúster, añada el objeto de tema como tema de clúster en el host de temas de clúster.
   - Si existe un objeto de tema con la misma serie de tema que el nombre del punto de suscripción, utilice el objeto de tema existente. Debe comprender las consecuencias del punto de suscripción que reutiliza un tema existente. Si el tema existente forma parte de una aplicación existente, debe resolver la colisión entre los dos temas con nombre idéntico.

- Si existe un objeto de tema con el mismo nombre que un punto de suscripción, pero con una serie de tema distinta, cree un tema con un nombre distinto.

3. Establezca el atributo **Topic** para WILDCARD en el valor BLOCK.

   El bloqueo de suscripciones a # o * aísla las suscripciones de comodín a puntos de suscripción, consulte Comodines y puntos de suscripción.
4. Establezca los atributos que requiera en el objeto de tema.

Ejemplo

El ejemplo muestra un archivo de mandato **runmqsc** que añade dos puntos de suscripción, USD y GBP.

```
DEFINE TOPIC(USD) TOPICSTR(USD)
DEFINE TOPIC(GBP) TOPICSTR(GBP) WILDCARD(BLOCK)
ALTER NL(SYSTEM.QPUBSUB.SUBPOINT.NAMELIST) NAMES(SYSTEM.BROKER.DEFAULT.SUBPOINT, USD, GBP)
```

Nota:

1. Incluya el punto de suscripción predeterminado a la lista de puntos de suscripción mediante el mandato **ALTER**. **ALTER** elimina los nombres existentes en la lista de nombres.
2. Defina los temas antes de modificar la lista de nombres. El gestor de colas sólo comprueba la lista de nombres cuando el gestor de colas se inicia y cuando la lista de nombres se modifica.

Configuración de redes de publicación/suscripción distribuidas

Los gestores de colas que están conectados entre sí en una topología de publicación/suscripción distribuida comparten un espacio de tema federado común. Las suscripciones creadas en un gestor de colas pueden recibir mensajes publicados por una aplicación conectada a otro gestor de colas de la topología.

Puede controlar el alcance de espacios de temas creados al conectar gestores de colas conjuntamente en clústeres o jerarquías. En un clúster de publicación/suscripción, un objeto de tema debe estar 'agrupado en clúster' para cada rama del espacio de temas que va a incluir el clúster. En una jerarquía, cada gestor de colas debe estar configurado para identificar su gestor de colas 'padre' en la jerarquía.

Puede controlar aún más el flujo de publicaciones y suscripciones dentro de la topología seleccionando si cada publicación y suscripción es local o global. Las publicaciones y suscripciones locales no se propagan más allá del gestor de colas al que el publicador o suscriptor está conectado.

Conceptos relacionados

Redes de publicación/suscripción distribuidas

Ámbito de la publicación

Ámbito de la suscripción

Espacios de temas

Tareas relacionadas

Definición de temas de clúster

Configurar un clúster de publicación/suscripción

Defina un tema en un gestor de colas. Para convertir el tema en un tema de clúster, establezca la propiedad **CLUSTER**. Para elegir el direccionamiento que se debe utilizar para publicaciones y suscripciones para este tema, establezca la propiedad **CLROUTE**.

Antes de empezar

Algunas configuraciones de clúster no pueden gestionar la sobrecarga de una publicación/suscripción direccionada directa. Antes de utilizar esta configuración, tenga en cuenta las consideraciones y opciones que se detallan en Diseño de clústeres de publicación/suscripción.

Para que los cambios realizados en un clúster se propaguen por todo el clúster, al menos un depósito completo debe estar siempre disponible. Asegúrese de que sus depósitos están disponibles antes de iniciar esta tarea.

Consulte también Direccionamiento para clústeres de publicación/suscripción: Notas sobre el comportamiento.

Escenario:

- El clúster de INVENTORY se ha configurado como se describe en “Añadir un gestor de colas a un clúster” en la página 335. Contiene tres gestores de colas; LONDON y NEWYORK tienen repositorios completos, PARIS contiene un repositorio parcial.

Acerca de esta tarea

Cuando define un tema en un gestor de colas en un clúster, debe especificar si el tema es un tema de clúster, y (si es así), el direccionamiento dentro del clúster para publicaciones y suscripciones de este tema. Para que el tema sea un tema de clúster, configure la propiedad **CLUSTER** en el objeto TOPIC con el nombre del clúster. Al definir un tema de clúster en un gestor de colas en el clúster, puede hacer que el tema esté disponible para todo el clúster. Para elegir el direccionamiento de mensajes que se va a utilizar dentro del clúster, establezca la propiedad **CLROUTE** en el objeto TOPIC en uno de los valores siguientes:

- **DIRECT**
- **TOPICHOST**

De forma predeterminada, el direccionamiento de temas es **DIRECT**. Cuando se configura un tema de clúster de direccionamiento directo en un gestor de colas, todos los gestores de colas del clúster reconocen los otros gestores de colas del clúster. Al realizar operaciones de publicación y suscripción, cada gestor de colas puede conectarse directamente a todos los otros gestores de colas del clúster. Consulte Clústeres de publicación/suscripción direccionados de forma directa.

A partir de IBM MQ 8.0, en su lugar, puede configurar el direccionamiento de temas como **TOPICHOST**. Cuando se utiliza el direccionamiento de host de temas, todos los gestores de colas del clúster pasan a reconocer los gestores de colas del clúster que alojan la definición del tema direccionado (es decir, los gestores de colas en los que se ha definido el objeto de tema). Cuando se realizan operaciones de publicación y suscripción, los gestores de colas del clúster sólo se conectan a estos gestores de colas de host de temas, no directamente entre sí. Los gestores de colas de host de temas son responsables del direccionamiento de publicaciones desde los gestores de colas en los que se publican publicaciones y los gestores de colas con suscripciones coincidentes. Consulte Clústeres de publicación/suscripción direccionados de host de tema.

Nota: Después de que un objeto de tema se haya agrupado (mediante la configuración del**CLUSTER** propiedad) no se puede cambiar el valor de la**CLROUTE** propiedad. El objeto se debe desagrupar del clúster (**CLUSTER** establecido en ' ') para poder cambiar el valor. Al desagrupar del clúster un tema, la definición de tema se convierte en un tema local, lo que produce un periodo durante el cual no se entregan publicaciones a las suscripciones de los gestores de colas remotos; esto se debe tener en cuenta al realizar este cambio. Consulte El efecto de definir un tema no de clúster con el mismo nombre que un tema de clúster de otro gestor de colas. Si intenta cambiar el valor del**CLROUTE** propiedad mientras está agrupada, el sistema genera una MQRCCF_CLROUTE_NOT_ALTERABLE excepción.

Procedimiento

1. Elija un gestor de colas para alojar el tema.

   Cualquier gestor de colas de clúster puede alojar un tema. Elija uno de los tres gestores de colas (LONDON, NEWYORK o PARIS) y configure las propiedades del objeto TOPIC. Si tiene previsto utilizar un direccionamiento directo, no hay ninguna diferencia operativa en relación con el gestor de colas que elija. Si tiene previsto utilizar el direccionamiento de host de tema, el gestor de colas seleccionado tiene responsabilidades adicionales para direccionar publicaciones. Por lo tanto, para el direccionamiento de host de tema, elija un gestor de colas que esté alojado en uno de los sistemas más potentes y con buena conectividad de red.

2. Defina un tema en un gestor de colas.

   Para que el tema sea un tema de clúster, incluya el nombre del clúster al definir el tema, y establezca el direccionamiento que prefiera utilizar para las publicaciones y suscripciones de este tema. Por ejemplo, para crear un tema de clúster de direccionamiento directo en el gestor de colas de LONDON, cree el tema de la siguiente manera:

   ```
   DEFINE TOPIC(INVENTORY) TOPICSTR('/INVENTORY') CLUSTER(INVENTORY) CLROUTE(DIRECT)
   ```

   Al definir un tema de clúster en un gestor de colas en el clúster, puede hacer que el tema esté disponible para todo el clúster.

   Para obtener más información sobre cómo utilizar **CLROUTE**, consulte DEFINE TOPIC (CLROUTE) y Direccionamiento para clústeres de publicación/suscripción: Notas sobre el comportamiento.

Resultados

El clúster está preparado para recibir publicaciones y suscripciones del tema.

Qué hacer a continuación

Si ha configurado un clúster de publicación/suscripción de direccionamiento directo, probablemente desee añadir un segundo host de tema para este tema. Consulte “Adición de hosts de temas adicionales a un clúster de direccionamiento de host de tema” en la página 464.

Si tiene varios clústeres de publicación/suscripción independientes, por ejemplo, debido a que su organización está geográficamente dispersa, puede propagar algunos temas de clúster en todos los clústeres. Para ello, conecte los clústeres en jerarquía. Consulte “Combinación de los espacios de temas de varios clústeres” en la página 469. También puede controlar el flujo de publicaciones de un clúster a otro. Consulte “Combinación y aislamiento de espacios de temas en varios clústeres” en la página 471.

Conceptos relacionados

Combinación de ámbitos de publicación y suscripción
A partir de IBM WebSphere MQ 7.0 en adelante, el ámbito de publicación y el de suscripción funcionan de forma independiente para determinar el flujo de las publicaciones entre gestores de cola.

Combinación de espacios de temas en redes de publicación/suscripción
Combine el espacio de temas de un gestor de colas con otros gestores de colas en un clúster o una jerarquía de publicación/suscripción. Combine los clústeres de publicación/suscripción, y los clústeres de publicación/suscripción con jerarquías.

Tareas relacionadas

Mover una definición de tema de clúster a un gestor de colas diferente
Para los clústeres de direccionamiento de host de tema o de direccionamiento directo, es posible que necesite mover una definición de tema de clúster cuando anula un gestor de colas o cuando un gestor de colas del clúster falla o no está disponible durante un periodo de tiempo prolongado.

Adición de hosts de temas adicionales a un clúster de direccionamiento de host de tema
En un clúster de publicación/suscripción de direccionamiento de host de tema, se pueden utilizar varios gestores de colas para direccionar publicaciones a las suscripciones definiendo el mismo objeto de clúster en estos gestores de colas. Esto se puede utilizar para mejorar la disponibilidad y equilibrar la carga de trabajo. Al añadir un host de tema adicional para el mismo objeto de tema de clúster, puede utilizar el parámetro **PUB** para controlar cuándo empiezan a direccionarse las publicaciones a través del nuevo host de tema.

Conexión de un gestor de colas a una jerarquía de publicación/suscripción
Puede conectar el gestor de colas hijo al gestor de colas padre de la jerarquía. Si el gestor de colas hijo ya es miembro de otra jerarquía o clúster, entonces esta conexión enlaza las jerarquías entre sí o enlaza el clúster a la jerarquía.

Desconexión de un gestor de colas de una jerarquía de publicación/suscripción

Desconecte un gestor de colas hijo de un gestor de colas padre en una jerarquía de publicación/suscripción.

Diseño de clústeres de publicación/suscripción
Resolución de problemas de publicación/suscripción distribuida
Inhabilitación de la publicación/suscripción en un clúster

Mover una definición de tema de clúster a un gestor de colas diferente

Para los clústeres de direccionamiento de host de tema o de direccionamiento directo, es posible que necesite mover una definición de tema de clúster cuando anula un gestor de colas o cuando un gestor de colas del clúster falla o no está disponible durante un periodo de tiempo prolongado.

Acerca de esta tarea

Puede tener varias definiciones del mismo objeto de tema de clúster en el clúster. Este es el estado normal de un clúster de direccionamiento de host de tema y no es el estado habitual de un clúster de direccionamiento directo. Para obtener más información, consulte Varias definiciones de tema de clúster del mismo nombre.

Para mover una definición de tema de clúster a un gestor de colas diferente en el clúster sin interrumpir el flujo de publicaciones, debe realizar estos pasos. El procedimiento mueve una definición del gestor de colas QM1 al gestor de colas QM2.

Procedimiento

1. Cree un duplicado de la definición de tema del clúster en QM2.

   Para el direccionamiento directo, establezca todos los atributos de modo que coincidan con la definición de QM1.

   Para el direccionamiento de host de tema, defina inicialmente el nuevo host de tema como PUB(DISABLED). De este modo, QM2 puede obtener información acerca de las suscripciones del clúster pero sin iniciar el direccionamiento de las publicaciones.
2. Espere hasta que se propague la información a través del clúster.

   Espere a que los gestores de colas de repositorio completo propaguen la nueva definición de tema de clúster a todos los gestores de colas del clúster. Utilice el mandato **DISPLAY CLUSTER** para visualizar los temas de clúster en cada miembro de clúster y compruebe si existe una definición cuyo origen sea QM2.

   En el direccionamiento de host de tema, espere a que el nuevo host de tema de QM2 obtenga información de todas las suscripciones. Compare las suscripciones del proxy conocidas para QM2 con las conocidas para QM1. Un modo de ver las suscripciones de proxy en un gestor de colas es emitir el mandato **runmqsc** siguiente:

   ```
   DISPLAY SUB(*) SUBTYPE(PROXY)
   ```
3. Para el direccionamiento de host de tema, vuelva a definir el host de tema en QM2 como PUB(ENABLED) y, a continuación, vuelva a definir el host de tema en QM1 como PUB(DISABLED).

   Ahora que el nuevo host de tema en QM2 está informado de todas las suscripciones que existen en los otros gestores de colas, el host de tema puede iniciar el direccionamiento de publicaciones.

   Si utiliza el valor PUB(DISABLED) para poner en pausa el tráfico de mensajes a través de QM1, asegúrese de que ninguna publicación esté en tránsito a través de QM1 cuando suprima la definición de tema de clúster.
4. Suprima la definición de tema de clúster de QM1.

   Solo puede suprimir la definición de QM1 si el gestor de colas está disponible. De lo contrario, debe realizar la ejecución con las dos definiciones existentes hasta que se reinicie QM1 o se fuerce su supresión.

Si QM1 continúa sin estar disponible durante un largo periodo de tiempo y, durante dicho periodo de tiempo, necesita modificar la definición de tema de clúster en QM2, la definición de QM2 será más reciente que la definición de QM1 y, por lo tanto, es la que suele prevalecer.

Durante este periodo, si existen diferencias entre las definiciones en QM1 y en QM2, los errores se graban en los registros de errores de los gestores de colas, indicándole la definición de tema de clúster que está en conflicto.

Si QM1 no regresará nunca al clúster, por ejemplo, debido a que se ha anulado de forma imprevista después de un error de hardware, como último recurso puede utilizar el mandato RESET CLUSTER para forzar la expulsión del gestor de colas. **RESET CLUSTER** suprime automáticamente todos los objetos de tema alojados en el gestor de colas de destino.

Conceptos relacionados

Combinación de ámbitos de publicación y suscripción
A partir de IBM WebSphere MQ 7.0 en adelante, el ámbito de publicación y el de suscripción funcionan de forma independiente para determinar el flujo de las publicaciones entre gestores de cola.

Combinación de espacios de temas en redes de publicación/suscripción
Combine el espacio de temas de un gestor de colas con otros gestores de colas en un clúster o una jerarquía de publicación/suscripción. Combine los clústeres de publicación/suscripción, y los clústeres de publicación/suscripción con jerarquías.

Tareas relacionadas

Configurar un clúster de publicación/suscripción
Defina un tema en un gestor de colas. Para convertir el tema en un tema de clúster, establezca la propiedad **CLUSTER**. Para elegir el direccionamiento que se debe utilizar para publicaciones y suscripciones para este tema, establezca la propiedad **CLROUTE**.

Adición de hosts de temas adicionales a un clúster de direccionamiento de host de tema
En un clúster de publicación/suscripción de direccionamiento de host de tema, se pueden utilizar varios gestores de colas para direccionar publicaciones a las suscripciones definiendo el mismo objeto de clúster en estos gestores de colas. Esto se puede utilizar para mejorar la disponibilidad y equilibrar la carga de trabajo. Al añadir un host de tema adicional para el mismo objeto de tema de clúster, puede utilizar el parámetro **PUB** para controlar cuándo empiezan a direccionarse las publicaciones a través del nuevo host de tema.

Conexión de un gestor de colas a una jerarquía de publicación/suscripción
Puede conectar el gestor de colas hijo al gestor de colas padre de la jerarquía. Si el gestor de colas hijo ya es miembro de otra jerarquía o clúster, entonces esta conexión enlaza las jerarquías entre sí o enlaza el clúster a la jerarquía.

Desconexión de un gestor de colas de una jerarquía de publicación/suscripción
Desconecte un gestor de colas hijo de un gestor de colas padre en una jerarquía de publicación/suscripción.

Adición de hosts de temas adicionales a un clúster de direccionamiento de host de tema

En un clúster de publicación/suscripción de direccionamiento de host de tema, se pueden utilizar varios gestores de colas para direccionar publicaciones a las suscripciones definiendo el mismo objeto de clúster en estos gestores de colas. Esto se puede utilizar para mejorar la disponibilidad y equilibrar la carga de trabajo. Al añadir un host de tema adicional para el mismo objeto de tema de clúster, puede utilizar el parámetro **PUB** para controlar cuándo empiezan a direccionarse las publicaciones a través del nuevo host de tema.

Antes de empezar

Definir el mismo objeto de tema de clúster en varios gestores de colas solo resulta funcionalmente útil para un clúster de direccionamiento de host de tema. Si se definen varios temas coincidente en un clúster de direccionamiento directo no se cambia su comportamiento. Esta tarea solo se aplica a los clústeres de direccionamiento de host de tema.

Esta tarea presupone que ha leído el artículo Varias definiciones de tema de clúster del mismo nombre, en especial las secciones siguientes:

- Varias definiciones de tema de clúster en un clúster de direccionamiento de host de temas
- Manejo especial para el parámetro PUB

Acerca de esta tarea

Cuando un gestor de colas se convierte en un host de tema de direccionamiento, en primer lugar, obtiene información de todos los temas relacionados que se han suscrito en el clúster. Si se está iniciando la publicación de publicaciones sobre estos temas en el momento en que se añade un host de tema adicional, y se direcciona una publicación al nuevo host de tema antes de que éste haya obtenido información acerca de la existencia de las suscripciones en otros gestores de colas del clúster, el nuevo host no reenvía dicha publicación a estas suscripciones. Esto hace que las suscripciones no obtengan las publicaciones.

Las publicaciones no se direccionan a través de los gestores de colas de host de tema que han establecido explícitamente el parámetro de objeto de clúster **PUB** en ENABLED, por lo que puede utilizar este valor para asegurarse de que de que a ninguna suscripción le falten publicaciones durante el proceso de adición de un host de tema adicional.

Nota: Mientras que un gestor de colas aloja un tema de clúster que se ha definido como PUB(DISABLED), los publicadores conectados a dicho gestor de colas no pueden publicar mensajes y las suscripciones coincidentes en dicho gestor de colas no reciben publicaciones publicadas en otros gestores de colas del clúster. Por este motivo, debe considerarse detenidamente la definición de temas de direccionamiento de host de tema en los gestores de colas donde existen suscripciones a los que se conectan las aplicaciones de publicación.

Procedimiento

1. Configure un nuevo host de tema y defina inicialmente el nuevo host de tema como PUB(DISABLED).

   De este modo, el nuevo host de tema puede obtener información acerca de las suscripciones del clúster pero sin iniciar el direccionamiento de las publicaciones.

   Para obtener información acerca de cómo configurar un host de tema, consulte la sección “Configurar un clúster de publicación/suscripción” en la página 460.
2. Determine cuándo el nuevo host de tema ha obtenido información acerca de todas las suscripciones.

   Para ello, compare las suscripciones del proxy conocidas para el nuevo host de tema con las conocidas para el host de tema existente. Una forma para ver las suscripciones de proxy es emitir el mandato **runmqsc** siguiente: DISPLAY SUB(*) SUBTYPE(PROXY)
3. Vuelva a definir el nuevo host de tema como PUB(ENABLED).

   Una vez que el nuevo host de tema está informado de todas las suscripciones existentes en otros gestores de colas, el tema puede iniciar el direccionamiento de publicaciones.

Conceptos relacionados

Combinación de ámbitos de publicación y suscripción
A partir de IBM WebSphere MQ 7.0 en adelante, el ámbito de publicación y el de suscripción funcionan de forma independiente para determinar el flujo de las publicaciones entre gestores de cola.

Combinación de espacios de temas en redes de publicación/suscripción
Combine el espacio de temas de un gestor de colas con otros gestores de colas en un clúster o una jerarquía de publicación/suscripción. Combine los clústeres de publicación/suscripción, y los clústeres de publicación/suscripción con jerarquías.

Tareas relacionadas

Configurar un clúster de publicación/suscripción
Defina un tema en un gestor de colas. Para convertir el tema en un tema de clúster, establezca la propiedad **CLUSTER**. Para elegir el direccionamiento que se debe utilizar para publicaciones y suscripciones para este tema, establezca la propiedad **CLROUTE**.

Mover una definición de tema de clúster a un gestor de colas diferente
Para los clústeres de direccionamiento de host de tema o de direccionamiento directo, es posible que necesite mover una definición de tema de clúster cuando anula un gestor de colas o cuando un gestor de colas del clúster falla o no está disponible durante un periodo de tiempo prolongado.

Conexión de un gestor de colas a una jerarquía de publicación/suscripción
Puede conectar el gestor de colas hijo al gestor de colas padre de la jerarquía. Si el gestor de colas hijo ya es miembro de otra jerarquía o clúster, entonces esta conexión enlaza las jerarquías entre sí o enlaza el clúster a la jerarquía.

Desconexión de un gestor de colas de una jerarquía de publicación/suscripción
Desconecte un gestor de colas hijo de un gestor de colas padre en una jerarquía de publicación/suscripción.

Combinación de ámbitos de publicación y suscripción

A partir de IBM WebSphere MQ 7.0 en adelante, el ámbito de publicación y el de suscripción funcionan de forma independiente para determinar el flujo de las publicaciones entre gestores de cola.

Las publicaciones pueden fluir a todos los gestores de colas conectados en una topología de publicación/suscripción, o sólo al gestor de colas local. El caso de las suscripciones proxy es similar. La combinación de estos dos flujos rige cuáles son las publicaciones que coinciden con una suscripción.

Tanto las publicaciones como las suscripciones pueden tener como ámbito `QMGR` o `ALL`. Si un publicador y un suscriptor están conectados al mismo gestor de colas, los valores de ámbito no afectan a las publicaciones que el suscriptor recibe del publicador.

Si el publicador y el suscriptor se conectan a gestores de colas diferentes, ambos valores deben ser `ALL` para recibir publicaciones remotas.

Supongamos que los publicadores están conectados a gestores de colas diferentes. Si desea que un suscriptor reciba publicaciones de cualquier publicador, establezca el ámbito de la suscripción en `ALL`. Entonces puede decidir, para cada publicador, si se limita el ámbito de sus publicaciones a suscriptores locales para el publicador.

Supongamos que los suscriptores están conectados a gestores de colas diferentes. Si desea que las publicaciones de un publicador se envíen a todos los suscriptores, establezca el ámbito de publicación en `ALL`. Si desea que un suscriptor reciba publicaciones sólo de un publicador conectado al mismo gestor de colas, establezca el ámbito de la suscripción en `QMGR`.

Ejemplo: servicio de resultados de fútbol

Suponga que es un equipo de una liga de fútbol. Cada equipo tiene un gestor de colas conectado a todos los otros equipos en un clúster de publicación/suscripción.

Los equipos publican los resultados de todos los partidos jugados en su campo utilizando el tema `Football/result/`*`Home team name/Away team name`*. Las series en cursiva son nombres de tema variable y la publicación es el resultado de la coincidencia.

Cada club también vuelve a publicar los resultados solo para el club usando la serie de tema `Football/myteam/`*`Home team name/Away team name`*.

Ambos temas se publican en todo el clúster.

Las suscripciones siguientes se han configurado por la liga para que los aficionados de un equipo puedan suscribirse a los resultados de tres maneras interesantes.

Observe que puede configurar temas de clúster con `SUBSCOPE(QMGR)`. Las definiciones de tema se propagan a cada miembro del clúster, pero el ámbito de la suscripción es sólo el gestor de colas local. De este modo, los suscriptores en cada gestor de colas reciben publicaciones diferentes de la misma suscripción.

Recibir todos los resultados

```
DEFINE TOPIC(A) TOPICSTR('Football/result/') CLUSTER SUBSCOPE(ALL)
```

Recibir todos los resultados locales

```
DEFINE TOPIC(B) TOPICSTR('Football/result/') CLUSTER SUBSCOPE(QMGR)
```

Dado que la suscripción tiene el ámbito QMGR, sólo coinciden los resultados publicados como equipo local.

Recibir todos los resultados de mi equipo

```
DEFINE TOPIC(C) TOPICSTR('Football/myteam/') CLUSTER SUBSCOPE(QMGR)
```

Dado que la suscripción tiene el ámbito QMGR, sólo coinciden los resultados del equipo local, que se vuelven a publicar localmente.

Conceptos relacionados

Combinación de espacios de temas en redes de publicación/suscripción
Combine el espacio de temas de un gestor de colas con otros gestores de colas en un clúster o una jerarquía de publicación/suscripción. Combine los clústeres de publicación/suscripción, y los clústeres de publicación/suscripción con jerarquías.

Redes de publicación/suscripción distribuidas

Ámbito de la publicación

Ámbito de la suscripción

Tareas relacionadas

Configurar un clúster de publicación/suscripción
Defina un tema en un gestor de colas. Para convertir el tema en un tema de clúster, establezca la propiedad **CLUSTER**. Para elegir el direccionamiento que se debe utilizar para publicaciones y suscripciones para este tema, establezca la propiedad **CLROUTE**.

Mover una definición de tema de clúster a un gestor de colas diferente
Para los clústeres de direccionamiento de host de tema o de direccionamiento directo, es posible que necesite mover una definición de tema de clúster cuando anula un gestor de colas o cuando un gestor de colas del clúster falla o no está disponible durante un periodo de tiempo prolongado.

Adición de hosts de temas adicionales a un clúster de direccionamiento de host de tema
En un clúster de publicación/suscripción de direccionamiento de host de tema, se pueden utilizar varios gestores de colas para direccionar publicaciones a las suscripciones definiendo el mismo objeto de clúster en estos gestores de colas. Esto se puede utilizar para mejorar la disponibilidad y equilibrar la carga de trabajo. Al añadir un host de tema adicional para el mismo objeto de tema de clúster, puede utilizar el parámetro **PUB** para controlar cuándo empiezan a direccionarse las publicaciones a través del nuevo host de tema.

Conexión de un gestor de colas a una jerarquía de publicación/suscripción
Puede conectar el gestor de colas hijo al gestor de colas padre de la jerarquía. Si el gestor de colas hijo ya es miembro de otra jerarquía o clúster, entonces esta conexión enlaza las jerarquías entre sí o enlaza el clúster a la jerarquía.

Desconexión de un gestor de colas de una jerarquía de publicación/suscripción
Desconecte un gestor de colas hijo de un gestor de colas padre en una jerarquía de publicación/suscripción.

Combinación de espacios de temas en redes de publicación/suscripción

Combine el espacio de temas de un gestor de colas con otros gestores de colas en un clúster o una jerarquía de publicación/suscripción. Combine los clústeres de publicación/suscripción, y los clústeres de publicación/suscripción con jerarquías.

Puede crear diferentes espacios de temas de publicación/suscripción utilizando los bloques de creación de los atributos **CLUSTER**, **PUBSCOPE** y **SUBSCOPE**, los clústeres de publicación/suscripción y las jerarquías de publicación/suscripción.

Empezando por el ejemplo de escalar de un único gestor de colas a un clúster de publicación/suscripción, los siguientes escenarios ilustran distintas topologías de publicación/suscripción.

Conceptos relacionados

Combinación de ámbitos de publicación y suscripción
A partir de IBM WebSphere MQ 7.0 en adelante, el ámbito de publicación y el de suscripción funcionan de forma independiente para determinar el flujo de las publicaciones entre gestores de cola.

Redes de publicación/suscripción distribuidas

Espacios de temas

Tareas relacionadas

Configurar un clúster de publicación/suscripción
Defina un tema en un gestor de colas. Para convertir el tema en un tema de clúster, establezca la propiedad **CLUSTER**. Para elegir el direccionamiento que se debe utilizar para publicaciones y suscripciones para este tema, establezca la propiedad **CLROUTE**.

Mover una definición de tema de clúster a un gestor de colas diferente
Para los clústeres de direccionamiento de host de tema o de direccionamiento directo, es posible que necesite mover una definición de tema de clúster cuando anula un gestor de colas o cuando un gestor de colas del clúster falla o no está disponible durante un periodo de tiempo prolongado.

Adición de hosts de temas adicionales a un clúster de direccionamiento de host de tema
En un clúster de publicación/suscripción de direccionamiento de host de tema, se pueden utilizar varios gestores de colas para direccionar publicaciones a las suscripciones definiendo el mismo objeto de clúster en estos gestores de colas. Esto se puede utilizar para mejorar la disponibilidad y equilibrar la carga de trabajo. Al añadir un host de tema adicional para el mismo objeto de tema de clúster, puede utilizar el parámetro **PUB** para controlar cuándo empiezan a direccionarse las publicaciones a través del nuevo host de tema.

Conexión de un gestor de colas a una jerarquía de publicación/suscripción
Puede conectar el gestor de colas hijo al gestor de colas padre de la jerarquía. Si el gestor de colas hijo ya es miembro de otra jerarquía o clúster, entonces esta conexión enlaza las jerarquías entre sí o enlaza el clúster a la jerarquía.

Desconexión de un gestor de colas de una jerarquía de publicación/suscripción
Desconecte un gestor de colas hijo de un gestor de colas padre en una jerarquía de publicación/suscripción.

Definición de temas de clúster

Creación de un solo espacio de tema en un clúster de publicación/suscripción

Aumentar un sistema de publicación/suscripción para ejecutarlo en varios gestores de colas. Utilice un clúster de publicación/suscripción para proporcionar a cada publicador y suscriptor un único espacio de tema idéntico.

Antes de empezar

Ha implementado un sistema de publicación/suscripción en un solo gestor de colas versión 7.

Cree siempre espacios de temas con sus propios temas raíz, en lugar de confiar en heredar los atributos de `SYSTEM.BASE.TOPIC`. Si aumenta el sistema de publicación/suscripción a un clúster, puede definir los temas raíz como temas de clúster, en el host de temas de clúster y, a continuación, todos los temas se compartirán en todo el clúster.

Acerca de esta tarea

Ahora desea aumentar el sistema para dar soporte a más publicadores y suscriptores y tienen todo el tema visible en todo el clúster.

Procedimiento

1. Cree un clúster para utilizarlo con el sistema de publicación/suscripción.

   Si tiene un clúster tradicional existente, por razones de rendimiento es mejor configurar un clúster nuevo para el nuevo sistema de suscripción a publicaciones. Puede utilizar los mismos servidores para los repositorios de clúster de ambos clústeres
2. Elija un gestor de colas, posiblemente uno de los repositorios, para ser el host de temas de clúster.
3. Asegúrese de que cada tema que se va a ver en todo el clúster de publicación/suscripción se resolverá en un objeto de tema administrativo.
   Establezca el atributo **CLUSTER** que nombra el clúster de publicación/suscripción.

Qué hacer a continuación

Conecte las aplicaciones de publicador y suscriptor a cualquier gestor de colas del clúster.

Cree objetos de tema administrativo que tengan el atributo **CLUSTER**. Los temas también se propagan por todo el clúster. Los programas de publicador y suscriptor utilizan los temas administrativos de modo que su comportamiento no se modifica por conectarse a diferentes gestores de colas del clúster

Si necesita `SYSTEM.BASE.TOPIC` para actuar como un tema de clúster en cada gestor de colas, debe modificarlo en cada gestor de colas.

Conceptos relacionados

Redes de publicación/suscripción distribuidas

Espacios de temas

Tareas relacionadas

Combinación de los espacios de temas de varios clústeres
Crear espacios de temas que abarquen varios clústeres. Publique en un tema de un clúster y suscríbase en otro.

Combinación y aislamiento de espacios de temas en varios clústeres
Aislar algunos espacios de temas a un clúster específico y combinar otros espacios de temas para hacerlos accesibles en todos los clústeres conectados.

Publicación y suscripción a espacios de temas en varios clústeres
Publicar y suscribirse a temas en varios clústeres utilizando clústeres solapados. Puede utilizar esta técnica mientras los espacios de temas en los clústeres no se solapen.

Definición de temas de clúster

Combinación de los espacios de temas de varios clústeres

Crear espacios de temas que abarquen varios clústeres. Publique en un tema de un clúster y suscríbase en otro.

Antes de empezar

Esta tarea presupone que tiene clústeres de publicación/suscripción de direccionamiento directo existentes y desea propagar algunos temas del clúster a todos los clústeres.

Nota: No puede realizar esta tarea para los clústeres de publicación/suscripción de direccionamiento directo.

Acerca de esta tarea

Para propagar publicaciones de un clúster a otro, tiene que unir los clústeres entre sí en una jerarquía; consulte Figura 65 en la página 470. Las conexiones jerárquicas propagan suscripciones y publicaciones entre los gestores de colas conectados y los clústeres propagan temas de clúster dentro de cada clúster, pero no entre clústeres.

La combinación de estos dos mecanismos propaga los temas de clúster entre todos los clústeres. Debe repetir los las definiciones de tema de clúster en cada clúster.

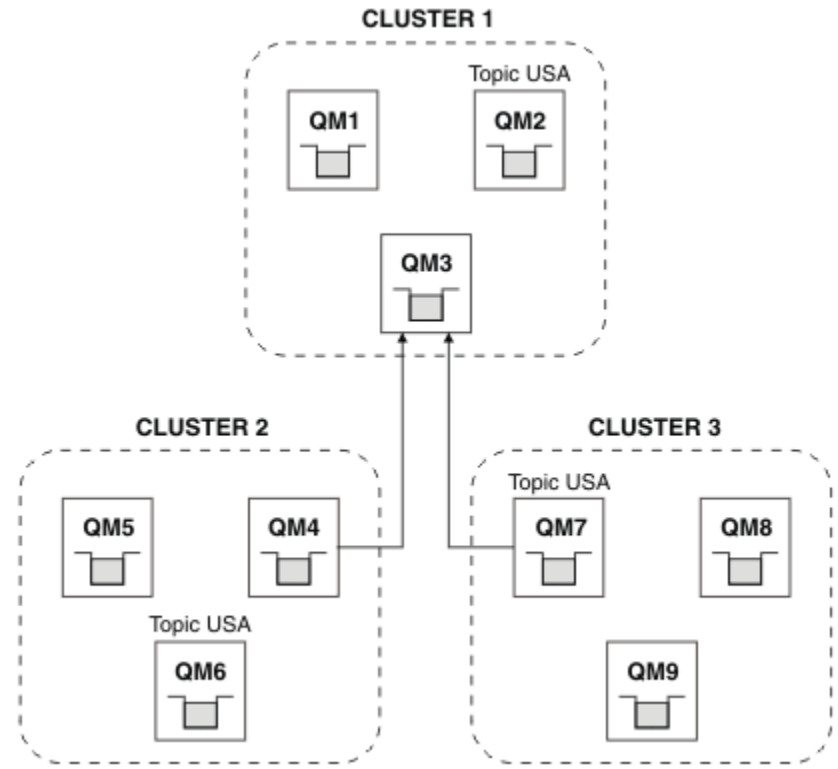


Figura 65. Conexión de clústeres mediante jerarquías

Los pasos siguientes conectan los clústeres en una jerarquía.

Procedimiento

1. Cree dos conjuntos de canales emisor-receptor para conectar QM3 y QM4, y QM3 y QM7, en ambas direcciones. Debe utilizar canales emisor-receptor y colas de transmisión tradicionales, en lugar de un clúster, para conectar una jerarquía.
2. Cree tres colas de transmisión con los nombres de los gestores de colas de destino. Utilice los alias de gestor de colas si, por alguna razón, no puede utilizar el nombre del gestor de colas de destino como el nombre de cola de transmisión.
3. Configure las colas de transmisión para desencadenar los canales emisores.
4. Compruebe que **PSMODE** de QM3, QM4 y QM7 se ha establecido en ENABLE.
5. Modifique el atributo **PARENT** de QM4 y QM7 a QM3.
6. Compruebe que el estado de la relación padre-hijo entre los gestores de colas está activo en ambas direcciones.
7. Cree el tema administrativo USA con el atributo **CLUSTER** ("CLUSTER 1"), **CLUSTER** ("CLUSTER 2") y **CLUSTER** ("CLUSTER 3") en cada uno de los tres gestores de colas de host de tema de clúster en los clústeres 1, 2 y 3. El host de tema de clúster no necesita ser un gestor de colas conectado de forma jerárquica.

Qué hacer a continuación

Ahora puede publicar en el tema de clúster USA en Figura 65 en la página 470 o suscribirse al mismo. Las suscripciones a publicaciones fluyen a publicadores y suscriptores en los tres clústeres.

Suponga que no ha creado USA como un tema de clúster en los otros clústeres. Si USA sólo se define en QM7, las publicaciones y suscripciones en USA se intercambian entre QM7, QM8, QM9 y QM3. Los publicadores y suscriptores que se ejecutan en QM7, QM8, QM9 heredan los atributos

del tema administrativo USA. Los publicadores y suscriptores de QM3 heredan los atributos de SYSTEM.BASE.TOPIC en QM3.

Consulte también “Combinación y aislamiento de espacios de temas en varios clústeres” en la página 471.

Conceptos relacionados

Redes de publicación/suscripción distribuidas

Espacios de temas

Tareas relacionadas

Creación de un solo espacio de tema en un clúster de publicación/suscripción
Aumentar un sistema de publicación/suscripción para ejecutarlo en varios gestores de colas. Utilice un clúster de publicación/suscripción para proporcionar a cada publicador y suscriptor un único espacio de tema idéntico.

Combinación y aislamiento de espacios de temas en varios clústeres
Aislar algunos espacios de temas a un clúster específico y combinar otros espacios de temas para hacerlos accesibles en todos los clústeres conectados.

Publicación y suscripción a espacios de temas en varios clústeres
Publicar y suscribirse a temas en varios clústeres utilizando clústeres solapados. Puede utilizar esta técnica mientras los espacios de temas en los clústeres no se solapen.

Definición de temas de clúster

Combinación y aislamiento de espacios de temas en varios clústeres

Aislar algunos espacios de temas a un clúster específico y combinar otros espacios de temas para hacerlos accesibles en todos los clústeres conectados.

Antes de empezar

Examine el tema “Combinación de los espacios de temas de varios clústeres” en la página 469. Es posible que sea suficiente para sus necesidades, sin añadir un gestor de colas adicional como puente.

Nota: Solo puede completar esta tarea utilizando clústeres de publicación/suscripción de direccionamiento directo. No puede realizar esta tarea utilizando clústeres de publicación/suscripción de direccionamiento directo.

Acerca de esta tarea

Una mejora potencial en la topología que se muestra en la Figura 65 en la página 470 de “Combinación de los espacios de temas de varios clústeres” en la página 469 es aislar temas de clúster que no se comparten en todos los clústeres. Aísle los clústeres creando un gestor de colas de puente que no esté en ninguno de los clústeres; consulte Figura 66 en la página 472. Utilice el gestor de colas de puente para filtrar cuáles son las publicaciones y suscripciones que pueden fluir de un clúster a otro.

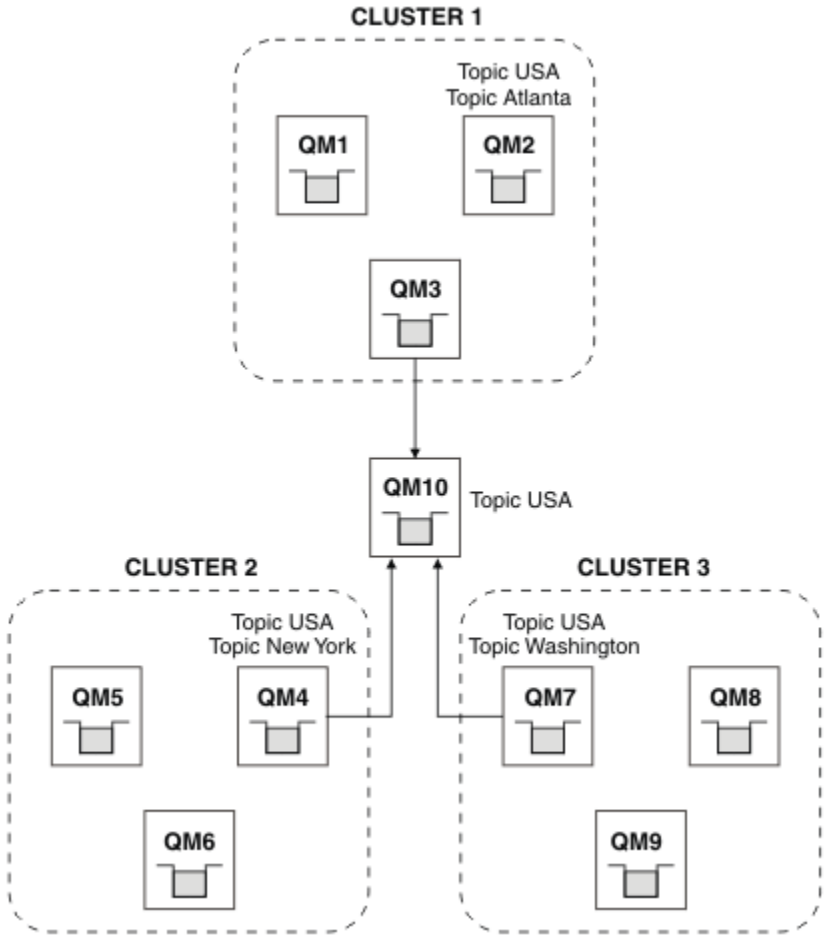


Figura 66. Clústeres de puente

Utilice el puente para aislar temas de clúster que no desee dejar expuestos en el puente en los demás clústeres. En Figura 66 en la página 472, USA es un tema de clúster compartido en todos los clústeres, y `Atlanta`, `New York` y `Washington` son temas de clúster que se comparten sólo en un clúster cada uno.

Modele la configuración mediante el procedimiento siguiente:

Procedimiento

1. Modifique todos los objetos de tema de `SYSTEM.BASE.TOPIC` para tener **SUBSCOPE** (`QMGR`) y **PUBSCOPE** (`QMGR`) en todos los gestores de colas.

   No se propaga ningún tema (ni siquiera temas de clúster) a otros gestores de colas a menos que se establezca **SUBSCOPE** (`ALL`) y **PUBSCOPE** (`ALL`) de forma explícita en el tema raíz de los temas del clúster.
2. Defina los temas en los tres gestores de colas de host de temas de clúster que desea compartir en cada clúster con los atributos **CLUSTER** (*clustername*), **SUBSCOPE** (`ALL`) y **PUBSCOPE** (`ALL`).

   Si desea que se compartan algunos temas de clúster entre todos los clústeres, defina el mismo tema en cada uno de los clústeres. Utilice el nombre de clúster de cada clúster como atributo del clúster.
3. Para los temas de clúster que desea compartir entre todos los clústeres, defina de nuevo los temas en el gestor de colas de puente (`QM10`), con los atributos **SUBSCOPE** (`ALL`) y **PUBSCOPE** (`ALL`).

Ejemplo

En el ejemplo de Figura 66 en la página 472, sólo los temas que heredan de USA se propagan entre los tres clústeres.

Qué hacer a continuación

Suscripciones para temas definidos en el gestor de colas de puente con **SUBSCOPE** (ALL) y **PUBSCOPE** (ALL) se propagan entre los clústeres.

Suscripciones para temas definidos en cada clúster con atributos **CLUSTER** (*clustername*), **SUBSCOPE** (ALL) y **PUBSCOPE** (ALL) se propagan dentro de cada clúster.

Las demás suscripciones son locales para un gestor de colas.

Conceptos relacionados

Redes de publicación/suscripción distribuidas

Espacios de temas

Ámbito de la publicación

Ámbito de la suscripción

Tareas relacionadas

Creación de un solo espacio de tema en un clúster de publicación/suscripción

Aumentar un sistema de publicación/suscripción para ejecutarlo en varios gestores de colas. Utilice un clúster de publicación/suscripción para proporcionar a cada publicador y suscriptor un único espacio de tema idéntico.

Combinación de los espacios de temas de varios clústeres

Crear espacios de temas que abarquen varios clústeres. Publique en un tema de un clúster y suscríbase en otro.

Publicación y suscripción a espacios de temas en varios clústeres

Publicar y suscribirse a temas en varios clústeres utilizando clústeres solapados. Puede utilizar esta técnica mientras los espacios de temas en los clústeres no se solapen.

Definición de temas de clúster

Publicación y suscripción a espacios de temas en varios clústeres

Publicar y suscribirse a temas en varios clústeres utilizando clústeres solapados. Puede utilizar esta técnica mientras los espacios de temas en los clústeres no se solapen.

Antes de empezar

Cree varios clústeres tradicionales con algunos gestores de colas en las intersecciones entre los clústeres.

Acerca de esta tarea

Es posible que haya elegido solapar clústeres por varias razones distintas.

1. Tiene un número limitado de servidores de alta disponibilidad, o gestores de colas. Puede decidir desplegar todos los repositorios del clúster y el tema de clúster los aloja.
2. Tiene clústeres de gestores de colas existentes que se conectan mediante gestores de colas de pasarela. Desea desplegar aplicaciones de publicación/suscripción a la misma topología del clúster.
3. Tiene varias aplicaciones de publicación/suscripción autocontenidas. Por razones de rendimiento, es mejor mantener los clústeres de publicación/suscripción pequeños y separados de los clústeres tradicionales. Ha decidido desplegar las aplicaciones en diferentes clústeres. No obstante, también desea supervisar todas las aplicaciones de publicación/suscripción en un gestor de colas, ya que sólo ha adquirido una licencia de la aplicación de supervisión. Este gestor de colas debe tener acceso a las publicaciones de temas de clúster en todos los clústeres.

Asegurándose de que los temas se definan en espacios de temas que no se solapen, puede desplegar los temas en clústeres de publicación/suscripción solapados, consulte Figura 67 en la página 474. Si los espacios de temas se solapan, desplegar en clústeres que se solapan ocasiona problemas.

Dado que los clústeres de publicación/suscripción se solapan, puede publicar y suscribirse a cualquiera de los espacios de temas utilizando los gestores de colas en el solapamiento.

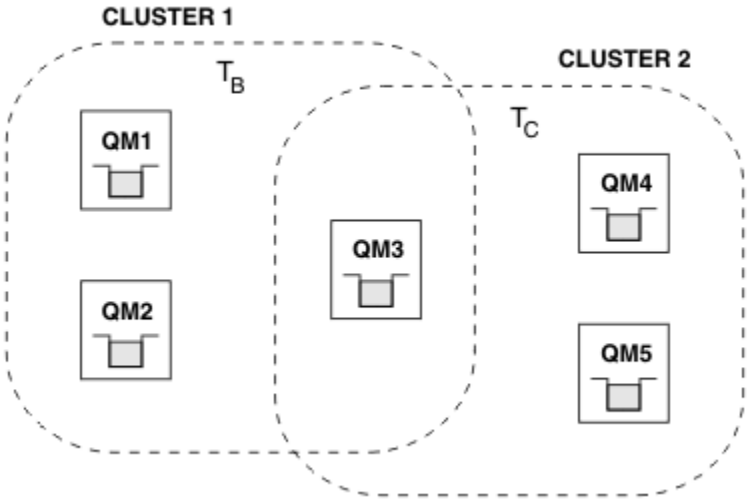


Figura 67. Clústeres que se solapan, espacios de temas que no se solapan

Procedimiento

Cree un medio de asegurar que los espacios de temas no se solapan.

Por ejemplo, defina un tema raíz exclusivo para cada uno de los espacios de temas. Convierta los temas raíz en temas de clúster.

a) `DEFINE TOPIC(B) TOPICSTR('B') CLUSTER('CLUSTER 1') ...`

b) `DEFINE TOPIC(C) TOPICSTR('C') CLUSTER('CLUSTER 2') ...`

Ejemplo

En Figura 67 en la página 474 los publicadores y suscriptores conectados a QM3 pueden publicar o suscribirse a T_B o T_C

Qué hacer a continuación

Conecte publicadores y suscriptores que utilicen temas en ambos clústeres para gestores de colas en el solapamiento.

Conecte publicadores y suscriptores que sólo deban utilizar temas en ambos clústeres para gestores de colas que no estén en el solapamiento.

Conceptos relacionados

Redes de publicación/suscripción distribuidas

Espacios de temas

Tareas relacionadas

Creación de un solo espacio de tema en un clúster de publicación/suscripción
Aumentar un sistema de publicación/suscripción para ejecutarlo en varios gestores de colas. Utilice un clúster de publicación/suscripción para proporcionar a cada publicador y suscriptor un único espacio de tema idéntico.

Combinación de los espacios de temas de varios clústeres
Crear espacios de temas que abarquen varios clústeres. Publique en un tema de un clúster y suscríbase en otro.

Combinación y aislamiento de espacios de temas en varios clústeres
Aislar algunos espacios de temas a un clúster específico y combinar otros espacios de temas para hacerlos accesibles en todos los clústeres conectados.

Definición de temas de clúster

Conexión de un gestor de colas a una jerarquía de publicación/suscripción

Puede conectar el gestor de colas hijo al gestor de colas padre de la jerarquía. Si el gestor de colas hijo ya es miembro de otra jerarquía o clúster, entonces esta conexión enlaza las jerarquías entre sí o enlaza el clúster a la jerarquía.

Antes de empezar

1. Los gestores de colas de una jerarquía de publicación/suscripción tienen nombres de gestor de colas exclusivos.
2. Una jerarquía de publicación/suscripción se basa en la característica del gestor de colas de "publicación/suscripción en cola". Ésta se debe habilitar en el gestor de colas padre y en el gestor de colas hijo. Consulte “Inicio de la publicación/suscripción en cola” en la página 456.
3. La relación de publicación/suscripción se basa en los canales emisor y receptor de la cola. Hay dos modos de establecer los canales:
   - Añadir el gestor de colas padre y el gestor de colas hijo a un clúster de IBM MQ. Consulte “Añadir un gestor de colas a un clúster” en la página 335.
   - Establecer un par de canales emisor/receptor desde el gestor de colas hijo al gestor de colas padre y desde el padre al hijo. Cada canal debe utilizar una cola de transmisión con el mismo nombre que el gestor de colas de destino, o un alias de gestor de colas con el mismo nombre que el gestor de colas de destino. Para obtener más información sobre cómo establecer una conexión de canal punto a punto, consulte “Técnicas de gestión de colas distribuidas de IBM MQ” en la página 211.

   Para obtener ejemplo sobre cómo configurar una jerarquía sobre cada tipo de configuración de canal, consulte el siguiente conjunto de escenarios de jerarquías de publicación/suscripción:

   - Escenario 1: utilización de canales de punto a punto con alias de nombre de gestor de colas
   - Escenario 2: utilización de canales de punto a punto con el mismo nombre para la cola de transmisión y el gestor de colas remoto
   - Escenario 3: utilización de un canal de clúster para añadir un gestor de colas

Acerca de esta tarea

Utilice el mandato ALTER QMGR PARENT (*PARENT_NAME*) **runmqsc** para conectar los hijos a los padres. Esta configuración se lleva a cabo en el gestor de colas hijo, donde *PARENT_NAME* es el nombre del gestor de colas padre.

Procedimiento

ALTER QMGR PARENT(*PARENT_NAME*)

Ejemplo

El primer ejemplo muestra cómo conectar el gestor de colas QM2 como hijo de QM1y, a continuación, consultar QM2 para confirmar que se ha convertido en un hijo con un **STATUS** de ACTIVO:

```
C:>runmqsc QM2
5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM2
alter qmgr parent(QM1)
     1 : alter qmgr parent(QM1)
AMQ8005: IBM MQ queue manager changed.
display pubsub all
     2 : display pubsub all
AMQ8723: Display pub/sub status details.
   QMNAME(QM2)                             TYPE(LOCAL)
   STATUS(ACTIVE)
AMQ8723: Display pub/sub status details.
   QMNAME(QM1)                             TYPE(PARENT)
STATUS(ACTIVE)
```

El siguiente ejemplo muestra el resultado de consultar a QM1 para estas conexiones:

```
C:\Documents and Settings\Admin>runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
display pubsub all
     2 : display pubsub all
AMQ8723: Display pub/sub status details.
   QMNAME(QM1)                             TYPE(LOCAL)
   STATUS(ACTIVE)
AMQ8723: Display pub/sub status details.
   QMNAME(QM2)                             TYPE(CHILD)
   STATUS(ACTIVE)
```

Si **STATUS** no se muestra como ACTIVE, compruebe que los canales entre el hijo y el padre estén bien configurados y se ejecuten correctamente. Compruebe los registros de errores de los dos gestores de colas para ver si hay errores.

Qué hacer a continuación

De forma predeterminada, los temas que utilizan los publicadores y suscriptores en un gestor de colas se comparten con los publicadores y suscriptores de los otros gestores de colas de la jerarquía. Los temas administrados se pueden configurar para controlar el nivel de compartición mediante el uso de las propiedades de tema **SUBSCOPE** y **PUBSCOPE**. Consulte “Configuración de redes de publicación/suscripción distribuidas” en la página 460

Conceptos relacionados

Combinación de ámbitos de publicación y suscripción

A partir de IBM WebSphere MQ 7.0 en adelante, el ámbito de publicación y el de suscripción funcionan de forma independiente para determinar el flujo de las publicaciones entre gestores de cola.

Combinación de espacios de temas en redes de publicación/suscripción

Combine el espacio de temas de un gestor de colas con otros gestores de colas en un clúster o una jerarquía de publicación/suscripción. Combine los clústeres de publicación/suscripción, y los clústeres de publicación/suscripción con jerarquías.

Tareas relacionadas

Configurar un clúster de publicación/suscripción

Defina un tema en un gestor de colas. Para convertir el tema en un tema de clúster, establezca la propiedad **CLUSTER**. Para elegir el direccionamiento que se debe utilizar para publicaciones y suscripciones para este tema, establezca la propiedad **CLROUTE**.

Mover una definición de tema de clúster a un gestor de colas diferente

Para los clústeres de direccionamiento de host de tema o de direccionamiento directo, es posible que necesite mover una definición de tema de clúster cuando anula un gestor de colas o cuando un gestor de colas del clúster falla o no está disponible durante un periodo de tiempo prolongado.

Adición de hosts de temas adicionales a un clúster de direccionamiento de host de tema

En un clúster de publicación/suscripción de direccionamiento de host de tema, se pueden utilizar varios gestores de colas para direccionar publicaciones a las suscripciones definiendo el mismo objeto de clúster en estos gestores de colas. Esto se puede utilizar para mejorar la disponibilidad y equilibrar la carga de trabajo. Al añadir un host de tema adicional para el mismo objeto de tema de clúster, puede utilizar el parámetro **PUB** para controlar cuándo empiezan a direccionarse las publicaciones a través del nuevo host de tema.

Desconexión de un gestor de colas de una jerarquía de publicación/suscripción

Desconecte un gestor de colas hijo de un gestor de colas padre en una jerarquía de publicación/suscripción.

Referencia relacionada

Corrientes y temas

DISPLAY PUBSUB

Mensajería de publicación/suscripción

Desconexión de un gestor de colas de una jerarquía de publicación/suscripción

Desconecte un gestor de colas hijo de un gestor de colas padre en una jerarquía de publicación/suscripción.

Acerca de esta tarea

Utilice el mandato **ALTER QMGR** para desconectar un gestor de colas de una jerarquía de intermediario. Puede desconectar un gestor de colas en cualquier orden y en cualquier momento.

La solicitud correspondiente para actualizar el padre se envía cuando la conexión entre los gestores de colas se está ejecutando.

Procedimiento

```
ALTER QMGR PARENT( '')
```

Ejemplo

```
C:\Documents and Settings\Admin>runmqsc QM2
5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM2.
    1 : alter qmgr parent('')
AMQ8005: IBM MQ queue manager changed.
    2 : display pubsub type(child)
AMQ8147: IBM MQ object  not found.
display pubsub type(parent)
    3 : display pubsub type(parent)
AMQ8147: IBM MQ object not found.
```

Qué hacer a continuación

Puede suprimir cualquier corriente, cola y canal definido manualmente que ya no se necesiten.

Conceptos relacionados

Combinación de ámbitos de publicación y suscripción
A partir de IBM WebSphere MQ 7.0 en adelante, el ámbito de publicación y el de suscripción funcionan de forma independiente para determinar el flujo de las publicaciones entre gestores de cola.

Combinación de espacios de temas en redes de publicación/suscripción
Combine el espacio de temas de un gestor de colas con otros gestores de colas en un clúster o una jerarquía de publicación/suscripción. Combine los clústeres de publicación/suscripción, y los clústeres de publicación/suscripción con jerarquías.

Tareas relacionadas

Configurar un clúster de publicación/suscripción
Defina un tema en un gestor de colas. Para convertir el tema en un tema de clúster, establezca la propiedad **CLUSTER**. Para elegir el direccionamiento que se debe utilizar para publicaciones y suscripciones para este tema, establezca la propiedad **CLROUTE**.

Mover una definición de tema de clúster a un gestor de colas diferente
Para los clústeres de direccionamiento de host de tema o de direccionamiento directo, es posible que necesite mover una definición de tema de clúster cuando anula un gestor de colas o cuando un gestor de colas del clúster falla o no está disponible durante un periodo de tiempo prolongado.

Adición de hosts de temas adicionales a un clúster de direccionamiento de host de tema
En un clúster de publicación/suscripción de direccionamiento de host de tema, se pueden utilizar varios gestores de colas para direccionar publicaciones a las suscripciones definiendo el mismo objeto de clúster en estos gestores de colas. Esto se puede utilizar para mejorar la disponibilidad y equilibrar la carga de trabajo. Al añadir un host de tema adicional para el mismo objeto de tema de clúster, puede utilizar el parámetro **PUB** para controlar cuándo empiezan a direccionarse las publicaciones a través del nuevo host de tema.

Conexión de un gestor de colas a una jerarquía de publicación/suscripción
Puede conectar el gestor de colas hijo al gestor de colas padre de la jerarquía. Si el gestor de colas hijo ya es miembro de otra jerarquía o clúster, entonces esta conexión enlaza las jerarquías entre sí o enlaza el clúster a la jerarquía.

ALW Configuración de varias instalaciones

Cuando se utilizan varias instalaciones en el mismo sistema, es necesario configurar las instalaciones y los gestores de colas.

Acerca de esta tarea

Esta información se aplica a AIX, Linux, and Windows.

Procedimiento

- Utilice la información de los siguientes enlaces para configurar las instalaciones:
  - “Modificación de la instalación principal” en la página 486
  - “Asociación de un gestor de colas con una instalación” en la página 487
  - “Conexión de aplicaciones en un entorno de varias instalaciones” en la página 478

ALW Conexión de aplicaciones en un entorno de varias instalaciones

En sistemas AIX, Linux, and Windows , si se cargan bibliotecas IBM MQ , IBM MQ utiliza automáticamente las bibliotecas adecuadas sin necesidad de realizar ninguna otra acción. IBM MQ utiliza bibliotecas de la instalación asociada al gestor de colas al que se conecta la aplicación.

Los conceptos siguientes se utilizan para explicar la forma en que las aplicaciones se conectan a IBM MQ:

Enlace
Cuando la aplicación se compila, la aplicación se enlaza a las bibliotecas de IBM MQ para obtener la función de las exportaciones que se cargan cuando se ejecuta la aplicación.

Cargando
Cuando se ejecuta la aplicación, se localizan y se cargan las bibliotecas de IBM MQ. El mecanismo específico que se utiliza para localizar las bibliotecas varía según el sistema operativo y según cómo se crea la aplicación. Para obtener más información sobre cómo localizar y cargar bibliotecas en un entorno de varias instalaciones, consulte “Carga de bibliotecas de IBM MQ” en la página 479.

Conectando
Cuando la aplicación se conecta a un gestor de colas en ejecución, por ejemplo mediante una llamada `MQCONN` o `MQCONNX`, se conecta utilizando las bibliotecas de IBM MQ cargadas.

Cuando una aplicación de servidor se conecta a un gestor de colas, las bibliotecas cargadas deben provenir de la instalación asociada con el gestor de colas. Con varias instalaciones en un sistema, esta restricción introduce nuevos cambios a la hora de seleccionar el mecanismo que el sistema operativo utiliza para localizar las bibliotecas de IBM MQ que hay que cargar:

- Cuando se emite el mandato **`setmqm`** para cambiar la instalación asociada con un gestor de colas, las bibliotecas que deben cargarse cambian.
- Cuando una aplicación se conecta a varios gestores de colas que son propiedad de distintas instalaciones, deben cargarse varios conjuntos de bibliotecas.

Sin embargo, si IBM MQ, las bibliotecas, se encuentran y se cargan, IBM MQ carga y utiliza las bibliotecas adecuadas sin que sea necesario realizar ninguna otra acción. Cuando la aplicación se conecta a un gestor de colas, IBM MQ carga las bibliotecas de la instalación con la que está asociado el gestor de colas.

Los escenarios de migración y la conexión de aplicaciones con varias instalaciones se considera más detalladamente en Coexistencia de gestores de colas de varias instalaciones en AIX, Linux, and Windows.

Para obtener más información sobre cómo cargar bibliotecas de IBM MQ, consulte “Carga de bibliotecas de IBM MQ” en la página 479.

Soporte y restricciones

Si alguna de las siguientes bibliotecas de IBM MQ se encuentra y se carga, el producto puede cargar y utilizar automáticamente las bibliotecas adecuadas:

- Bibliotecas de servidor C
- Bibliotecas de servidor C++
- Bibliotecas de servidor XA
- Bibliotecas de servidor COBOL
- Bibliotecas de servidor COM++
- .NET en modalidad no gestionada

IBM MQ también carga y utiliza automáticamente las bibliotecas apropiadas para aplicaciones Java y JMS en modalidad de enlaces.

Existe una serie de restricciones para aplicaciones que utilizan varias instalaciones. Para obtener más información, consulte “Restricciones para aplicaciones que utilizan varias instalaciones” en la página 482.

Conceptos relacionados

“Restricciones para aplicaciones que utilizan varias instalaciones” en la página 482
Existen restricciones cuando se utilizan bibliotecas de servidor CICS, conexiones de vía rápida, manejadores de mensajes y salidas en un entorno de varias instalaciones.

“Carga de bibliotecas de IBM MQ” en la página 479
Al decidir cómo cargar las bibliotecas de IBM MQ, debe tener en cuenta varios factores, entre ellos: el entorno, si se pueden cambiar las aplicaciones existentes, si desea una instalación principal, dónde está instalado IBM MQ y si es probable que cambie la ubicación de IBM MQ.

Tareas relacionadas

Elección de una instalación primaria
“Modificación de la instalación principal” en la página 486
Puede utilizar el mandato **setmqinst** para establecer o anular una instalación como instalación principal.

“Asociación de un gestor de colas con una instalación” en la página 487
Cuando se crea un gestor de colas, éste se asocia automáticamente a la instalación que ha emitido el mandato **crtmqm**. En AIX, Linux, and Windows, puede cambiar la instalación asociada a un gestor de colas mediante el mandato **setmqm**.

ALW Carga de bibliotecas de IBM MQ

Al decidir cómo cargar las bibliotecas de IBM MQ, debe tener en cuenta varios factores, entre ellos: el entorno, si se pueden cambiar las aplicaciones existentes, si desea una instalación principal, dónde está instalado IBM MQ y si es probable que cambie la ubicación de IBM MQ.

El modo en que se localizan y se cargan las bibliotecas de IBM MQ depende del entorno de la instalación:

- En sistemas AIX and Linux , si una copia de una versión de IBM MQestá instalada en la ubicación predeterminada, las aplicaciones existentes siguen funcionando de la misma forma que las versiones anteriores. Sin embargo, si las aplicaciones necesitan enlaces simbólicos en `/usr/lib`, debe seleccionar una instalación de versión de IBM MQ para que sea la instalación primaria o crear manualmente los enlaces simbólicos.
- Si IBM MQ está instalado en una ubicación no predeterminada, es posible que tenga que cambiar las aplicaciones existentes para que se carguen las bibliotecas correctas.

El modo en que se localizan y se cargan las bibliotecas de IBM MQ también depende de cómo están configuradas las aplicaciones existentes para cargar aplicaciones. Para obtener más información sobre

cómo se pueden cargar las bibliotecas, consulte “Mecanismo de carga de bibliotecas del sistema operativo” en la página 481.

De manera óptima, debe asegurarse de que la biblioteca de IBM MQ, que carga el sistema operativo, es la biblioteca a la que está asociado el gestor de colas.

Los métodos para cargar las bibliotecas de IBM MQ varían según la plataforma, y cada método tiene sus ventajas y desventajas.

Tabla 29. Ventajas y desventajas de las opciones para cargar bibliotecas

Plataforma	Opción	Ventajas	Desventajas
Linux AIX Sistemas AIX and Linux	Establecer o cambiar la vía de búsqueda de ejecución incorporada (RPath) de la aplicación. Esta opción requiere que vuelva a compilar y enlazar la aplicación. Para obtener más información sobre cómo compilar y enlazar aplicaciones, consulte Creación de una aplicación de procedimientos.	• El ámbito del cambio es claro.	• Debe poder volver a compilar y enlazar la aplicación. • Si la ubicación de IBM MQ cambia, debe cambiar el valor de RPath.
Sistemas AIX and Linux	Establezca la variable de entorno *LD_LIBRARY_PATH*, utilizando setmqenv o crtmqenv, con la opción **-k** o **-1**. (AIX En AIX, esta variable de entorno es *LIBPATH*	• No es necesario realizar ningún cambio en las aplicaciones existentes. • Se sustituyen los valores de RPath incluidos en una aplicación. • Es fácil cambiar la variable si la ubicación de IBM MQ cambia.	• Las aplicaciones `setuid` y `setgid`, o las aplicaciones creadas de otros modos, es posible que ignoren *LD_LIBRARY_PATH* por motivos de seguridad. • Es específica del entorno, por lo que debe establecerse en cada entorno donde se ejecuta la aplicación. • Posible impacto en otras aplicaciones que se basan en *LD_LIBRARY_PATH*. • Linux Linux: El compilador utilizado para crear la aplicación puede inhabilitar el uso de *LD_LIBRARY_PATH*. Para obtener más información, consulte Consideraciones sobre el enlace en tiempo de ejecución para Linux .

Tabla 29. Ventajas y desventajas de las opciones para cargar bibliotecas (continuación)			
Plataforma	**Opción**	**Ventajas**	**Desventajas**
Windows Sistemas Windows	Establecer la variable PATH utilizando setmqenv o crtmqenv.	• No es necesario realizar ningún cambio en las aplicaciones existentes. • Es fácil cambiar la variable si la ubicación de IBM MQ cambia.	• Es específica del entorno, por lo que debe establecerse en cada entorno donde se ejecuta la aplicación. • Posible impacto en otras aplicaciones.
ALW Sistemas AIX, Linux, and Windows	Establecer la instalación principal en una instalación de IBM MQ o posterior. Consulte “Modificación de la instalación principal” en la página 486. Para obtener más información sobre la instalación principal, consulte Elección de una instalación principal.	• No es necesario realizar ningún cambio en las aplicaciones existentes. • Es fácil cambiar la instalación principal si la ubicación de IBM MQ cambia. • Ofrece un comportamiento similar al de versiones anteriores de IBM MQ.	• Linux AIX AIX and Linux: No funciona si `/usr/lib` no está en la vía de acceso de búsqueda predeterminada.

Consideraciones sobre la carga de bibliotecas para Linux

Linux

Las aplicaciones compiladas utilizando algunas versiones de gcc, por ejemplo, la versión 3.2.x, pueden tener una RPath incorporada que no se pueda sustituir utilizando la variable de entorno *LD_LIBRARY_PATH*. Puede determinar si una aplicación se ve afectada utilizando el mandato `readelf -d` *`applicationName`* . RPath no puede sustituirse si el símbolo RPATH está presente y el símbolo RUNPATH no está presente.

Mecanismo de carga de bibliotecas del sistema operativo

En sistemas Windows, se busca en varios directorios para encontrar las bibliotecas:

- El directorio desde el que se carga la aplicación.
- El directorio actual.
- Los directorios de la variable de entorno *PATH*, tanto la variable *PATH* como la variable *PATH* del usuario actual.

Linux AIX En sistemas AIX and Linux, hay diversos métodos que pueden haberse utilizado para localizar las bibliotecas que hay que cargar:

- Utilizando la variable de entorno *LD_LIBRARY_PATH* (también *LIBPATH* en AIX). Si esta variable está establecida, define un conjunto de directorios en los que se buscan las bibliotecas de IBM MQ necesarias. Si se encuentran bibliotecas en estos directorios, estas se utilizan preferentemente en vez de las bibliotecas que se puedan encontrar utilizando otros métodos.
- Utilizando una vía de acceso de búsqueda incorporada (RPath). La aplicación puede contener un conjunto de directorios en los que buscar bibliotecas de IBM MQ. Si la variable *LD_LIBRARY_PATH* no está establecida, o si las bibliotecas necesarias no se encuentran utilizando la variable, se realiza una búsqueda de las bibliotecas en la RPath. Si las aplicaciones existentes utilizan una RPath, pero no es posible volver a compilar y enlazar la aplicación, debe instalar IBM MQ en la ubicación predeterminada, o bien utilizar otro método para encontrar las bibliotecas.

- Mediante la vía de acceso de biblioteca predeterminada. Si las bibliotecas de IBM MQ no se encuentran después de realizar una búsqueda en la variable *LD_LIBRARY_PATH* y en las ubicaciones RPath, se realiza una búsqueda en la vía de acceso de bibliotecas predeterminada. Normalmente, esta vía de acceso contiene `/usr/lib` o `/usr/lib64`. Si las bibliotecas no se encuentran después de realizar una búsqueda en la vía de acceso de bibliotecas predeterminada, la aplicación no se puede iniciar porque faltan dependencias.

Puede utilizar los mecanismos del sistema operativo para averiguar si las aplicaciones tienen una vía de acceso de búsqueda incluida. Por ejemplo:

- AIX: **dump**
- Linux: **readelf**

Conceptos relacionados

"Restricciones para aplicaciones que utilizan varias instalaciones" en la página 482
Existen restricciones cuando se utilizan bibliotecas de servidor CICS, conexiones de vía rápida, manejadores de mensajes y salidas en un entorno de varias instalaciones.

"Conexión de aplicaciones en un entorno de varias instalaciones" en la página 478
En sistemas AIX, Linux, and Windows , si se cargan bibliotecas IBM MQ , IBM MQ utiliza automáticamente las bibliotecas adecuadas sin necesidad de realizar ninguna otra acción. IBM MQ utiliza bibliotecas de la instalación asociada al gestor de colas al que se conecta la aplicación.

Tareas relacionadas

Elección de una instalación primaria

"Modificación de la instalación principal" en la página 486
Puede utilizar el mandato **setmqinst** para establecer o anular una instalación como instalación principal.

"Asociación de un gestor de colas con una instalación" en la página 487
Cuando se crea un gestor de colas, éste se asocia automáticamente a la instalación que ha emitido el mandato **crtmqm**. En AIX, Linux, and Windows, puede cambiar la instalación asociada a un gestor de colas mediante el mandato **setmqm**.

ALW Restricciones para aplicaciones que utilizan varias instalaciones

Existen restricciones cuando se utilizan bibliotecas de servidor CICS, conexiones de vía rápida, manejadores de mensajes y salidas en un entorno de varias instalaciones.

Bibliotecas de servidor CICS

Si se utilizan las bibliotecas de servidor CICS, IBM MQ no selecciona automáticamente el nivel de biblioteca correcto. Debe compilar y enlazar sus aplicaciones con el nivel de biblioteca adecuado para el gestor de colas al que se conecta la aplicación. Para obtener más información, consulte Creación de bibliotecas para utilizarlas con TXSeries para Multiplatforms versión 5.

Manejadores de mensajes

Los manejadores de mensajes que utilizan el valor especial `MQHC_UNASSOCIATED_HCONN` están limitados a utilizar la primera instalación cargada en un proceso. Si el manejador de mensajes no puede utilizarse en una instalación determinada, se devuelve el código de razón `MQRC_HMSG_NOT_AVAILABLE`.

Esta restricción afecta a las propiedades de los mensajes. No se pueden utilizar manejadores de mensajes para obtener propiedades de mensaje de un gestor de colas de una instalación y colocarlas en un gestor de colas de otra instalación. Para obtener más información sobre los descriptores de contexto de mensaje, consulte MQCRTMH - Crear descriptor de contexto de mensaje..

Salidas

En un entorno de varias instalaciones, las salidas existentes deben actualizarse para utilizarlas con las instalaciones de IBM MQ . Las salidas de conversión de datos generadas mediante el mandato **crtmqcvx** deben generarse de nuevo mediante el mandato actualizado.

Todas las salidas deben escribirse utilizando la estructura MQIEP, no pueden utilizar una variable RPATH incluida para localizar las bibliotecas de IBM MQ y no se pueden enlazar con bibliotecas de IBM MQ. Para obtener más información, consulte Escritura de funciones de salida y servicios instalables en AIX, Linux, and Windows .

Vía rápida

En un servidor con varias instalaciones, las aplicaciones que utilizan una conexión de vía de acceso rápida a IBM MQ deben seguir estas reglas:

1. El gestor de colas debe estar asociado con la misma instalación desde la que la aplicación ha cargado las bibliotecas en tiempo de ejecución de IBM MQ. La aplicación o debe utilizar una conexión de vía rápida para un gestor de colas asociado con una instalación distinta. Un intento de hacer que la conexión resulte en un error y en el código de razón MQRC_INSTALLATION_MISMATCH.
2. La conexión de vía no rápida a un gestor de colas asociado con la misma instalación que el gestor desde el que la aplicación ha cargado las bibliotecas en tiempo de ejecución de IBM MQ impide que la aplicación se conecte por la vía rápida, a menos que se cumpla una de estas condiciones:
   - La aplicación realiza su primera conexión con un gestor de colas asociado con la misma instalación que una conexión de vía rápida.
   - La variable de entorno, AMQ_SINGLE_INSTALLATION está establecida.
3. La conexión de una vía de acceso no rápida a un gestor de colas asociado a una instalación de IBM MQ no tiene ningún efecto sobre si una aplicación puede conectarse a la vía de acceso rápida.

Con el conjunto AMQ_SINGLE_INSTALLATION, puede hacer que cualquier conexión con un gestor de colas sea una conexión de vía de acceso rápida. En caso contrario, se aplican casi las mismas restricciones:

- la instalación debe ser la misma que la instalación desde la que se han cargado las bibliotecas en tiempo de ejecución de IBM MQ.
- Cada conexión al mismo proceso debe ser a la misma instalación. Si intenta conectarse a un gestor de colas asociado con una instalación diferente, la conexión falla con el código de razón MQRC_INSTALLATION_MISMATCH. Tenga en cuenta que con AMQ_SINGLE_INSTALLATION establecido, esta restricción se aplica a todas las conexiones, no solo a las conexiones de vía de acceso rápida.
- Conecte sólo un gestor de colas con conexiones de vía rápida.

Referencia relacionada

MQCONNX - Conectar gestor de colas (ampliado)
Estructura MQIEP
2583 (0A17) (RC2583): MQRC_INSTALLATION_MISMATCH
2587 (0A1B) (RC2587): MQRC_HMSG_NOT_AVAILABLE
2590 (0A1E) (RC2590): MQRC_FASTPATH_NOT_AVAILABLE

ALW Conexión de aplicaciones .NET en un entorno de varias instalaciones

De forma predeterminada, las aplicaciones utilizan los ensamblajes .NET de la instalación principal. Si no existe ninguna instalación principal, o si no desea utilizar los ensamblajes de la instalación principal, debe actualizar el archivo de configuración de la aplicación o la variable de entorno *DEVPATH*.

Si existe una instalación principal en el sistema, los ensamblajes .NET y los archivos de políticas de esa instalación se registran en la caché de ensamblajes global (GAC). Los ensamblajes .NET para todas

las demás instalaciones pueden encontrarse en la vía de instalación de cada instalación, pero los ensamblajes no se registran en la GAC. Por consiguiente, de forma predeterminada, las aplicaciones se ejecutan utilizando los ensamblajes .NET de la instalación principal. Debe actualizar el archivo de configuración de la aplicación si se cumple alguna de estas condiciones:

- No tiene ninguna instalación principal.
- No desea que la aplicación utilice los ensamblajes de la instalación principal.
- La instalación principal corresponde a una versión de IBM MQ anterior a la versión con la que se ha compilado la aplicación.

Para obtener información sobre cómo actualizar el archivo de configuración de la aplicación, consulte "Conexión de aplicaciones .NET utilizando el archivo de configuración de aplicación" en la página 484.

Debe actualizar la variable de entorno *DEVPATH* si la siguiente afirmación es verdadera:

- Desea que su aplicación utilice los ensamblajes de una instalación no principal, pero la instalación principal corresponde a la misma versión que la instalación no principal.

Para obtener más información sobre cómo actualizar la variable *DEVPATH*, consulte "Conexión de aplicaciones .NET utilizando DEVPATH" en la página 485.

Conexión de aplicaciones .NET utilizando el archivo de configuración de aplicación

En el archivo de configuración de la aplicación, debe definir varios códigos para redireccionar aplicaciones para que utilicen ensamblajes que no corresponden a la instalación principal.

La siguiente tabla muestra los cambios específicos que deben realizarse en el archivo de configuración de la aplicación para permitir que las aplicaciones .NET se conecten utilizando ensamblajes determinados:

Tabla 30. Configuración de aplicaciones para utilizar ensamblajes determinados

	Aplicaciones compiladas con una versión anterior de IBM MQ	**Aplicaciones compiladas con una versión posterior de IBM MQ**
Para ejecutar una aplicación con una instalación principal de IBM MQ de una versión posterior. (ensamblajes de versión posterior en la GAC):	No es necesario realizar cambios	No es necesario realizar cambios
Para ejecutar una aplicación con una instalación principal de IBM MQ de una versión anterior. (ensamblajes de versión anterior en la GAC):	No es necesario realizar cambios	En el archivo de configuración de la aplicación: • Utilice el código *bindingRedirect* para indicar el uso de la versión anterior de los ensamblajes que se encuentran en la GAC
Para ejecutar una aplicación con una instalación no principal de IBM MQ de una versión posterior. (ensamblajes de versión posterior en la carpeta de instalación):	En el archivo de configuración de la aplicación: • Utilice el código *codebase* para que apunte a la ubicación de los ensamblajes de la versión posterior. • Utilice el código *bindingRedirect* para indicar el uso de los ensamblajes de la versión posterior.	En el archivo de configuración de la aplicación: • Utilice el código *codebase* para que apunte a la ubicación de los ensamblajes de la versión posterior.

Tabla 30. Configuración de aplicaciones para utilizar ensamblajes determinados (continuación)		
	Aplicaciones compiladas con una versión anterior de IBM MQ	**Aplicaciones compiladas con una versión posterior de IBM MQ**
Para ejecutar una aplicación con una versión anterior de la instalación no principal de IBM MQ. (ensamblajes de versión anterior en la carpeta de instalación):	En el archivo de configuración de la aplicación: • Utilice el código *codebase* para que apunte a la ubicación de los ensamblajes de la versión anterior. • Incluya el código *publisherpolicy Apply=no*	En el archivo de configuración de la aplicación: • Utilice el código *codebase* para que apunte a la ubicación de los ensamblajes de la versión anterior. • Utilice el código *bindingRedirect* para indicar el uso de los ensamblajes de la versión anterior. • Incluya el código *publisherpolicy Apply=no*

Se proporciona un archivo de configuración de la aplicación de ejemplo `NonPrimaryRedirect.config` en la carpeta *MQ_INSTALLATION_PATH*`\tools\dotnet\samples\base`. Este archivo se puede modificar con la vía de acceso de instalación de IBM MQ de cualquier instalación no principal. El archivo también puede incluirse directamente en otros archivos de configuración utilizando el código *linkedConfiguration*. Se proporcionan ejemplos para `nmqsget.exe.config` y `nmqsput.exe.config`. Ambos ejemplos utilizan el código *linkedConfiguration* e incluyen el archivo `NonPrimaryRedirect.config`.

Conexión de aplicaciones .NET utilizando DEVPATH

Puede encontrar los ensamblajes utilizando la variable de entorno *DEVPATH*. Los ensamblajes especificados mediante la variable *DEVPATH* se utilizan con preferencia sobre cualquier otro ensamblaje de la GAC. Consulte la documentación apropiada de Microsoft sobre *DEVPATH* para obtener más información sobre cuándo utilizar esta variable.

Para buscar los ensamblajes utilizando la variable de entorno *DEVPATH*, debe establecer la variable de entorno *DEVPATH* en la carpeta que contiene los ensamblajes que desea utilizar. A continuación, debe actualizar el archivo de configuración de la aplicación y añadir la siguiente información de configuración en tiempo de ejecución:

```
<configuration>
<runtime>
<developmentMode developerInstallation="true" />
</runtime>
</configuration>
```

Conceptos relacionados

“Conexión de aplicaciones en un entorno de varias instalaciones” en la página 478
En sistemas AIX, Linux, and Windows , si se cargan bibliotecas IBM MQ , IBM MQ utiliza automáticamente las bibliotecas adecuadas sin necesidad de realizar ninguna otra acción. IBM MQ utiliza bibliotecas de la instalación asociada al gestor de colas al que se conecta la aplicación.

Varias instalaciones

Tareas relacionadas

Elección de una instalación primaria

Utilización de .NET

ALW Modificación de la instalación principal

Puede utilizar el mandato **setmqinst** para establecer o anular una instalación como instalación principal.

Acerca de esta tarea

Esta tarea se aplica a AIX, Linux, and Windows.

La instalación principal es la instalación a la que hacen referencia ubicaciones necesarias definidas a nivel del sistema. Para obtener más información sobre la instalación principal y las consideraciones para elegir la instalación principal, consulte Elección de una instalación principal.

Windows Durante el proceso de instalación en Windows, puede especificar que la instalación sea la instalación principal.

Linux AIX En sistemas AIX and Linux, debe emitir el mandato **setmqinst** después de la instalación para establecer la instalación como instalación principal.

Procedimiento

- Para establecer una instalación como primaria, realice los pasos siguientes:
  a) Compruebe si una instalación ya se ha designado como instalación principal; para ello, escriba el mandato siguiente:

  ```
  MQ_INSTALLATION_PATH/bin/dspmqinst
  ```

  donde *MQ_INSTALLATION_PATH* es la vía de acceso de instalación de una instalación de IBM MQ .
  b) Si una instalación de IBM MQ existente se establece como instalación primaria, desestablézcala antes de continuar con el paso siguiente.
  c) Asegúrese de estar conectado con la autorización adecuada:
  – Linux AIX Como root en AIX and Linux.
  – Windows Como miembro del grupo de administradores en sistemas Windows.
  d) Ejecute uno de los mandatos siguientes:
  – Para establecer la instalación principal utilizando la vía de acceso de la instalación que desea designar como instalación principal:

  ```
  MQ_INSTALLATION_PATH/bin/setmqinst -i -p MQ_INSTALLATION_PATH
  ```

  – Para establecer la instalación principal utilizando el nombre de la instalación que desea designar como instalación principal:

  ```
  MQ_INSTALLATION_PATH/bin/setmqinst -i -n installationName
  ```

  e) Windows
  En sistemas Windows, reinicie el sistema.
- Para anular la designación de una instalación como primaria, realice los pasos siguientes:
  a) Compruebe qué instalación se ha designado como instalación principal; para ello, escriba el mandato siguiente:

  ```
  MQ_INSTALLATION_PATH/bin/dspmqinst
  ```

  donde *MQ_INSTALLATION_PATH* es la vía de acceso de instalación de una instalación de IBM MQ .

b) Asegúrese de estar conectado con la autorización adecuada:

- Linux AIX Como root en AIX and Linux.
- Windows Como miembro del grupo de administradores en sistemas Windows.

- Ejecute uno de los mandatos siguientes:
  - Para anular la designación de instalación principal utilizando la vía de acceso de la instalación cuya designación como instalación principal desea anular:

```
MQ_INSTALLATION_PATH/bin/setmqinst -x -p MQ_INSTALLATION_PATH
```

  - Para anular la designación de instalación principal utilizando el nombre de la instalación cuya designación como instalación principal desea anular:

```
MQ_INSTALLATION_PATH/bin/setmqinst -x -n installationName
```

Tareas relacionadas

Desinstalación, actualización y mantenimiento de la instalación principal

Elección de un nombre de instalación

Referencia relacionada

Características que solamente se pueden utilizar con la instalación en Windows

Enlaces a bibliotecas externas y mandatos de control para la instalación principal en AIX and Linux

setmqinst

ALW Asociación de un gestor de colas con una instalación

Cuando se crea un gestor de colas, éste se asocia automáticamente a la instalación que ha emitido el mandato **crtmqm**. En AIX, Linux, and Windows, puede cambiar la instalación asociada a un gestor de colas mediante el mandato **setmqm**.

Acerca de esta tarea

La instalación a la que un gestor de colas está asociado limita el gestor de colas, de modo que pueda administrarse sólo con mandatos de la instalación. Existen tres excepciones clave:

- **setmqm** cambia la instalación asociada al gestor de colas. Este mandato debe emitirse desde la instalación que desea asociar al gestor de colas, no desde la instalación a la que actualmente está asociado el gestor de colas. El nombre de instalación especificado por el mandato **setmqm** debe coincidir con la instalación desde la que se emite el mandato.
- **strmqm** debe emitirse desde la instalación asociada con el gestor de colas.
- **dspmq** muestra información sobre todos los gestores de colas de un sistema, no solamente de los gestores de colas asociados a la misma instalación que el mandato **dspmq**. El mandato `dspmq -o installation` muestra información sobre qué gestores de colas están asociados a cada instalación.

Para los entornos de alta disponibilidad, el mandato **addmqinf** asocia automáticamente el gestor de colas a la instalación desde la que se ha emitido el mandato **addmqinf**. Siempre que el mandato **strmqm** se emita desde la misma instalación que el mandato **addmqinf**, no es necesario llevar a cabo ningún otro tipo de configuración. Para iniciar el gestor de colas utilizando una instalación distinta, en primer lugar debe cambiar la instalación asociada mediante el mandato **setmqm**.

Si se desea asociar un gestor de colas a una instalación, se puede utilizar el mandato **setmqm** de las siguientes maneras:

- Mover gestores de colas individuales entre versiones equivalentes de IBM MQ. Por ejemplo, mover un gestor de colas de un sistema de prueba a uno de producción.

- Migrar gestores de colas individuales de una versión anterior de IBM MQ a una versión de IBM MQ más reciente. La migración entre versiones de gestores de colas tiene varias implicaciones que debe tener en cuenta. Para obtener más información sobre la migración, consulte Mantenimiento y migración.

Procedimiento

1. Detenga el gestor de colas ejecutando el mandato **endmqm** desde la instalación que está asociada actualmente al gestor de colas.
2. Asocie el gestor de colas a otra instalación mediante el mandato **setmqm** desde dicha instalación.

   Por ejemplo, para establecer que el gestor de colas QMB esté asociado a una instalación denominada Installation2, escriba el siguiente mandato desde Installation2:

   ```
   MQ_INSTALLATION_PATH/bin/setmqm -m QMB -n Installation2
   ```

   donde *MQ_INSTALLATION_PATH* es la vía de acceso en la que Installation2 se encuentra instalada.
3. Inicie el gestor de colas mediante el mandato **strmqm** desde la instalación que ahora está asociada al gestor de colas.

   Este mandato realiza la migración de gestor de colas necesaria y, como consecuencia de ello, el gestor de colas pasa a estar disponible para su uso.

Qué hacer a continuación

Si la instalación a la que está asociado un gestor de colas se ha suprimido, o si la información de estado del gestor de colas no está disponible, el mandato **setmqm** no puede asociar el gestor de colas a otra instalación. En esta situación, realice las siguientes acciones:

1. Utilice el mandato **dspmqinst** para ver las otras instalaciones del sistema.
2. Modifique manualmente el campo InstallationName de la stanza QueueManager del archivo mqs.ini para especificar otra instalación.
3. Utilice el mandato **dltmqm** desde dicha instalación para suprimir el gestor de colas.

Conceptos relacionados

“Búsqueda de instalaciones de IBM MQ en un sistema” en la página 488
Si tiene varias instalaciones de IBM MQ en un sistema, puede comprobar qué versiones están instaladas y dónde se encuentran.

“Archivo de configuración de IBM MQ, mqs.ini” en la página 96
El archivo de configuración IBM MQ, mqs.ini, contiene información relevante para todos los gestores de colas del nodo. Se crea automáticamente durante la instalación.

Tareas relacionadas

Elección de una instalación primaria

Referencia relacionada

addmqinf
dspmq
dspmqinst
endmqm
setmqm
strmqm

ALW Búsqueda de instalaciones de IBM MQ en un sistema

Si tiene varias instalaciones de IBM MQ en un sistema, puede comprobar qué versiones están instaladas y dónde se encuentran.

Puede utilizar los métodos siguientes para encontrar las instalaciones de IBM MQ existentes en el sistema:

- Utilice las herramientas de instalación de la plataforma para consultar dónde se ha instalado IBM MQ. A continuación, utilice el mandato **dspmqver** desde una instalación de IBM MQ . Los siguientes mandatos son ejemplos de mandatos que se pueden utilizar para consultar donde se ha instalado IBM MQ:
  - AIX En los sistemas AIX, puede utilizar el mandato **lslpp**:

    ```
    lslpp -R ALL -l mqm.base.runtime
    ```

  - Linux En los sistemas Linux, puede utilizar el mandato **rpm**:

    ```
    rpm -qa --qf "%{NAME}-%{VERSION}-%{RELEASE}\t%{INSTPREFIXES}\n" | grep MQSeriesRuntime
    ```

  - Windows En los sistemas Windows, puede utilizar el mandato **wmic**. Este mandato podría instalar el cliente wmic:

    ```
    wmic product where "(Name like '%MQ%') AND (not Name like '%bitSupport')" get Name,
    Version, InstallLocation
    ```

- Linux AIX En sistemas AIX and Linux, emita el mandato siguiente para averiguar dónde se ha instalado IBM MQ:

  ```
  cat /etc/opt/mqm/mqinst.ini
  ```

  A continuación, utilice el mandato **dspmqver** desde una instalación de IBM MQ .
- Windows Para visualizar los detalles de instalaciones en el sistema, en Windows de 32 bits, emita el mandato siguiente:

  ```
  reg.exe query "HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphere MQ\Installation" /s
  ```

- Windows En Windows de 64 bits, emita el siguiente mandato:

  ```
  reg.exe query "HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\IBM\WebSphere MQ\Installation" /s
  ```

Referencia relacionada
dspmqver
dspmqinst
Varias instalaciones

Configuración de la alta disponibilidad, la recuperación y el reinicio

Puede hacer que las aplicaciones tengan alta disponibilidad manteniendo la disponibilidad de las colas si un gestor de colas falla, y recuperando los mensajes tras una anomalía de almacenamiento o del servidor.

Acerca de esta tarea

z/OS En z/OS, la alta disponibilidad está integrada en la plataforma. Consulte Colas compartidas y grupos de compartición de colas.

Multi En Multiplatforms, puede mejorar la disponibilidad de las aplicaciones cliente mediante la reconexión de cliente para cambiar un cliente automáticamente entre un grupo de gestores de colas o con la nueva instancia de un gestor de colas multiinstancia después de que el gestor de colas falle. La reconexión automática de cliente no está soportada en IBM MQ classes for Java. Un gestor de colas multiinstancia se configura para ejecutarse como un único gestor de colas en varios servidores.

Las aplicaciones de servidor se despliegan en el gestor de colas. Si falla el servidor que ejecutando la instancia activa, la ejecución cambia automáticamente a una instancia en espera del mismo gestor de colas en un servidor diferente. Si configura las aplicaciones de servidor para que se ejecuten como servicios del gestor de colas, se reinician cuando la instancia en espera pasa a ser la instancia activa del gestor de colas.

Otra manera de aumentar la disponibilidad de las aplicaciones de servidor en Multiplatforms es desplegar las aplicaciones de servidor en varios sistemas en un clúster de gestor de colas. A partir de IBM WebSphere MQ 7.1, la recuperación de errores de clúster vuelve a ejecutar las operaciones que han causado problemas hasta que estos se resuelven. Consulte “Cambios en la recuperación de errores de clúster en servidores en Multiplatforms” en la página 714. También puede configurar IBM MQ for Multiplatforms como parte de la solución de agrupación en clúster específica de plataforma, como:

- Microsoft Servidor de clúster
- IBM i Clústeres HA en IBM i
- Linux AIX PowerHA para AIX (anteriormente HACMP en AIX) y otras soluciones en clúster de UNIX and Linux

Linux En los sistemas Linux, puede configurar gestores de colas de datos replicados (RDQM) para implementar soluciones de alta disponibilidad o de recuperación tras desastre. Para la alta disponibilidad, las instancias del mismo gestor de colas se configuran en cada nodo de un grupo de tres servidores Linux. Una de las tres instancias es la instancia activa. Los datos del gestor de colas activo se replican de forma síncrona hacia las otras dos instancias, para que una de dichas instancias pueda tomar el relevo en caso de producirse algún fallo. Para la recuperación tras desastre, un gestor de colas se ejecuta en un nodo primario en un sitio, con una instancia secundaria de ese gestor de colas ubicado en un nodo de recuperación en un sitio diferente. Los datos se replican entre la instancia primaria y la instancia secundaria, y si el nodo primario se pierde por cualquier motivo, la instancia secundaria puede convertirse en la instancia primaria e iniciarse.

CP4I HA nativa es una solución de alta disponibilidad destinada a los contenedores. HA nativa utiliza la réplica de registro para mantener actualizadas tres instancias de un gestor de colas que se ejecutan en nodos distintos. Una instancia está activa en cualquier momento y procesa los mensajes. El gestor de colas activo envía sus actualizaciones de registro a las otras dos instancias para mantenerlas actualizadas. Si la instancia activa falla, una de las instancias de réplica asume automáticamente el rol activo.

MQ Appliance Otra opción para una solución de alta disponibilidad o recuperación tras desastre es desplegar un par de dispositivos IBM MQ. Consulte Alta disponibilidad y Recuperación tras desastre en la documentación de IBM MQ Appliance.

Un sistema de mensajería asegura que los mensajes que han entrado en el sistema se entreguen en su destino. IBM MQ puede rastrear la ruta de un mensaje cuando se mueve de un gestor de colas a otro mediante el mandato **dspmqrte**. Si un sistema falla, los mensajes pueden recuperarse de varias formas según el tipo de anomalía y la forma en la que esté configurado el sistema. IBM MQ mantiene registros de recuperación de las actividades de los gestores de colas que gestionan la recepción, la transmisión y la entrega de mensajes.Utiliza estos registros para tres tipos de recuperación:

1. *Recuperación de reinicio*, cuando se detiene IBM MQ de forma planificada.
2. *Recuperación de anomalía*, cuando una anomalía detiene IBM MQ.
3. *Recuperación desde medio de almacenamiento*, para restaurar objetos dañados.

En todos los casos, la recuperación restaura el gestor de colas y lo devuelve al estado en el que estaba cuando se detuvo, pero las transacciones que estaban en curso se restituyen y se eliminan de las colas todas las actualizaciones que estaban en curso en el momento en que se detuvo el gestor de colas. La recuperación restaura todos los mensajes persistentes; los mensajes no persistentes pueden perderse durante el proceso.

PRECAUCIÓN: No puede mover los registros de recuperación a un sistema operativo distinto.

Reconexión de cliente automática

Puede hacer que las aplicaciones cliente se reconecten automáticamente, sin tener que escribir código adicional, configurando una serie de componentes.

La reconexión de cliente automática es *en línea*. La conexión se restaura automáticamente en cualquier punto del programa de aplicación cliente y se restauran todos los manejadores para abrir objetos.

Por el contrario, la reconexión manual necesita que la aplicación cliente vuelva a crear una conexión mediante `MQCONN` o `MQCONNX` y que vuelva a abrir los objetos. La reconexión de cliente automática es adecuada para muchas aplicaciones cliente, pero no para todas.

En la Tabla 31 en la página 491 se muestra el release más antiguo de soporte del cliente de IBM MQ que se debe instalar en una estación de trabajo de cliente. Debe actualizar las estaciones de trabajo de cliente a uno de estos niveles para que una aplicación pueda utilizar la reconexión automática de cliente. La Tabla 32 en la página 492 lista otros requisitos para habilitar la reconexión automática de cliente.

Mediante el acceso de programa a las opciones de reconexión, una aplicación cliente puede establecer las opciones de reconexión. Salvo los clientes JMS y XMS, si una aplicación cliente tiene acceso a las opciones de reconexión, también puede crear un manejador de sucesos para manejar sucesos de reconexión.

Una aplicación cliente existente puede ser capaz de beneficiarse del soporte de la reconexión, sin recompilar ni enlazar:

- Para un cliente que no seaJMS, establezca la variable de entorno `mqclient.ini` `DefRecon` para establecer las opciones de reconexión. Utilice una CCDT para conectarse a un gestor de colas. Si el cliente va a conectarse a un gestor de colas multiinstancia, proporcione las direcciones de red de las instancias de gestor de colas activa y de reserva en la CCDT. Para un gestor de colas de datos duplicados, o un gestor de colas HA en un IBM MQ Appliance, puede especificar una dirección IP flotante utilizada por ambos gestores de colas, el activo y el en espera, para simplificar la configuración.
- Para un cliente JMS, establezca las opciones de reconexión en la configuración de la fábrica de conexiones. Cuando se ejecutan dentro del contenedor EJB de un servidor Java EE, los beans controlados por mensaje (MDB) se pueden reconectar a IBM MQ utilizando el mecanismo de reconexión que proporcionan las especificaciones de activación del adaptador de recursos de IBM MQ (o los puertos de escucha si se ejecutan en WebSphere Application Server). Sin embargo, si la aplicación no es un MDB (o si se ejecuta en el contenedor web), la aplicación debe implementar su propia lógica de reconexión ya que la reconexión automática del cliente no está soportada en este caso. El adaptador de recursos de IBM MQ proporciona esta capacidad de reconexión para la entrega de mensajes a beans controlados por mensajes, pero otros elementos de Java EE como los servlets deben implementar su propia reconexión.

Nota: IBM MQ classes for Java no da soporte a la reconexión automática del cliente.

Tabla 31. Clientes soportados

Interfaz del cliente	**Cliente**	**Acceso de programa a las opciones de reconexión**	**Soporte de reconexión**
API de mensajería	C, C++, COBOL, Visual Basic no gestionado, XMS (XMS no gestionado en Windows)	7.0.1	7.0.1
	JMS (JSE, contenedor de cliente y contenedores gestionados Java EE)	7.0.1.3	7.0.1.3
	IBM MQ classes for Java	No soportado	No soportado
	Clientes XMS gestionado y .NET gestionado: C#, Visual Basic,	7.1	7.1

Tabla 31. Clientes soportados (continuación)			
Interfaz del cliente	**Cliente**	**Acceso de programa a las opciones de reconexión**	**Soporte de reconexión**
Otras API	Windows Communication Foundation (no gestionado [1])	No soportado	7.0.1
	Windows Communication Foundation (gestionado [1])	No soportado	No soportado
	Axis 1	No soportado	No soportado
	Axis 2	No soportado	7.0.1.3
	HTTP (web 2.0)	No soportado	7.0.1.3

1. Establezca la modalidad gestionada o no gestionada en la configuración de enlace WCF.

La reconexión automática tiene los siguientes requisitos de configuración:

Tabla 32. Requisitos de configuración de reconexión automática		
Componente	**Requisitos de configuración de reconexión automática**	**Efecto del incumplimiento del requisito**
Instalación de IBM MQ MQI client	Consulte Tabla 31 en la página 491	`MQRC_OPTIONS_ERROR`
Instalación de servidor de IBM MQ	Nivel 7.0.1	`MQRC_OPTIONS_ERROR`
Canal	`SHARECNV > 0`	`MQRC_ENVIRONMENT_ERROR`
Entorno de aplicaciones	Debe ser con hebras	`MQRC_ENVIRONMENT_ERROR`
MQI	Uno de los siguientes: • `MQCONNX` con las `opciones MQCNO` establecidas en `MQCNO_RECONNECT` o en `MQCNO_RECONNECT_Q_MGR`. • `Defrecon=YES|QMGR` en `mqclient.ini` • En JMS, establezca la propiedad **`CLIENTRECONNECTOPTIONS`** de la fábrica de conexiones.	`MQCC_FAILED` cuando se interrumpe una conexión o el gestor de colas finaliza o falla.

La Figura 68 en la página 493 muestra las principales interacciones entre componentes que están involucrados en la reconexión de cliente.

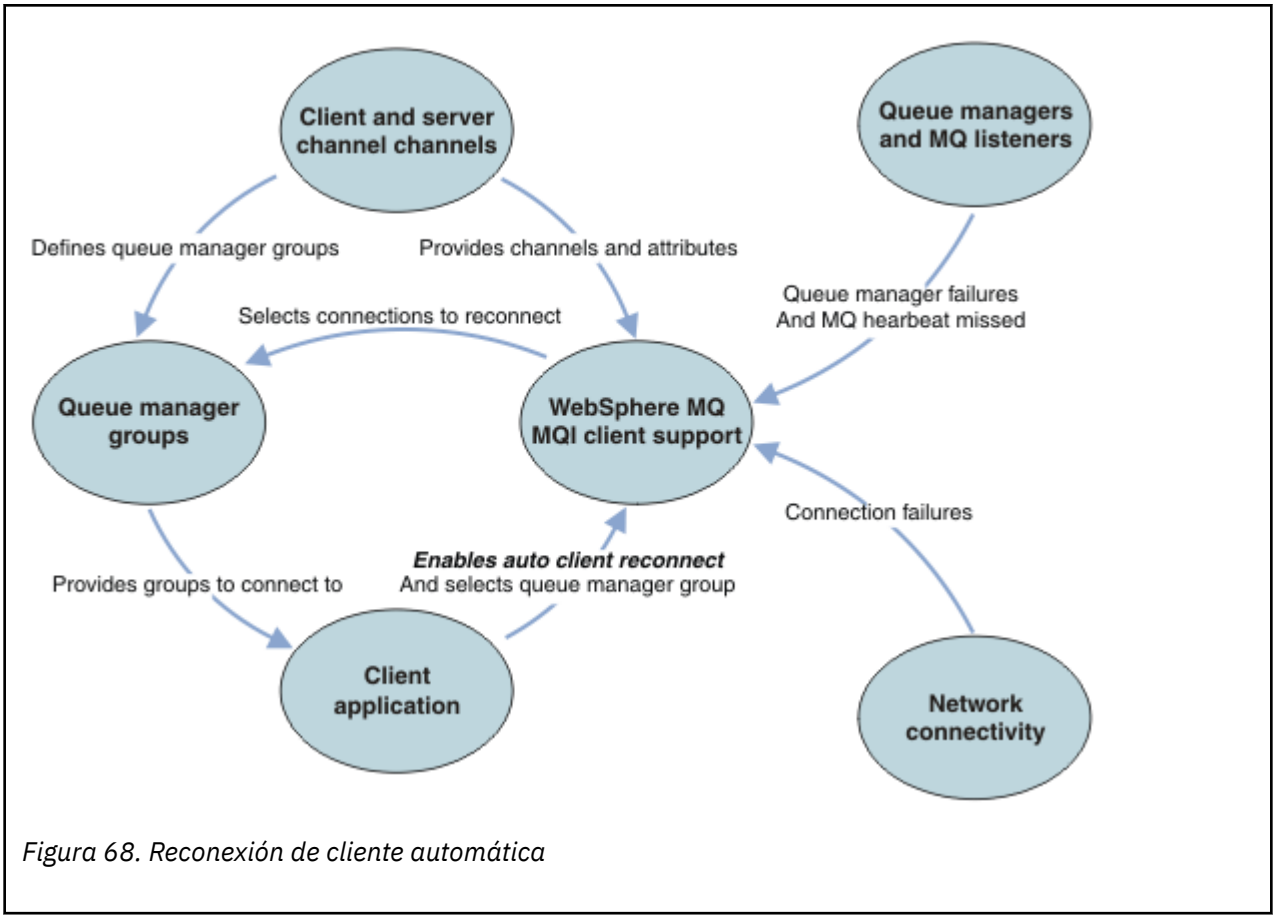


Figura 68. Reconexión de cliente automática

Aplicación de cliente

La aplicación cliente es un IBM MQ MQI client. Para obtener detalles sobre la reconexión de cliente automática para un cliente JMS, consulte Utilización de la reconexión automática de cliente de JMS.

- De forma predeterminada, los clientes no se vuelven a conectar automáticamente. Habilite la reconexión automática estableciendo la opción `MQCONNX` `MQCNO` `MQCNO_RECONNECT` o `MQCNO_RECONNECT_Q_MGR`.
- Muchas aplicaciones están escritas de forma que puedan aprovechan la reconexión automática sin código adicional. Habilite la reconexión automática para los programas existentes, sin realizar ningún cambio de codificación, estableciendo el atributo `DefRecon` en la stanza de canales del archivo de configuración de `mqclient.ini`.
- Utilice una de estas tres opciones:

  1. Modificar el programa para que la reconexión no afecte a la lógica. Por ejemplo, es posible que tenga que emitir llamadas MQI dentro del punto de sincronización y volver a enviar las transacciones restituidas. Los consumidores asíncronos deben comprobar si se han 'suspendido' si se restituye una transacción.
  2. Añadir un manejador de sucesos para detectar la reconexión y restablecer el estado de la aplicación cliente cuando la conexión se restablece.
  3. No habilitar la reconexión automática: en cambio, desconecte el cliente y emita una llamada `MQCONN` o `MQCONNX` MQI para buscar otra instancia del gestor de colas que se ejecuta en el mismo grupo del gestor de colas.

  Para obtener más detalles sobre estas tres opciones, consulte “Recuperación de la aplicación” en la página 588.
- La reconexión a un gestor de colas con el mismo nombre no garantiza la reconexión a la misma instancia de un gestor de colas.

Utilice una opción MQCNO MQCNO_RECONNECT_Q_MGRpara volver a conectarse a una instancia del mismo gestor de colas.

- Un cliente puede registrar un manejador de sucesos de manera que pueda ser informado sobre el estado de reconexión. El MQHCONN pasado en el manejador de sucesos no se puede utilizar. Se proporcionan los códigos de razón siguientes:

  MQRC_RECONNECTING
  La conexión ha fallado, y el sistema está intentando volverse a conectar. Recibirá varios sucesos de MQRC_RECONNECTING si se realizan varios intentos de reconexión.

  MQRC_RECONNECTED
  Se ha realizado reconexión, y todos los manejadores se han restablecido de forma satisfactoria.

  MQRC_RECONNECT_FAILED
  La reconexión no ha resultado satisfactoria.

  MQRC_RECONNECT_QMID_MISMATCH
  Una conexión reconectable ha especificado MQCNO_RECONNECT_Q_MGR y la conexión ha intentado volver a conectarse a un gestor de colas diferente.

  MQRC_RECONNECT_Q_MGR_REQD
  En el programa cliente se ha especificado una opción como, por ejemplo, MQMO_MATCH_MSG_TOKEN en una llamada MQGET, que requiere la reconexión con el mismo gestor de colas.

- Un cliente reconectable está habilitado para volverse a conectar de forma automática solo después de conectarse. Es decir, la llamada MQCONNX en sí misma no se reintenta si falla. Por ejemplo, si recibe el código de retorno 2543 - MQRC_STANDBY_Q_MGR de MQCONNX, deberá volver a emitir la llamada tras un breve periodo.

  MQRC_RECONNECT_INCOMPATIBLE
  Se devuelve este código de razón cuando la aplicación intenta utilizar MQPMO_LOGICAL_ORDER (con MQPUT y MQPUT1) o MQGMO_LOGICAL_ORDER (con MQGET) cuando se establecen las opciones de reconexión. El motivo por el cual se devuelve el código de razón es garantizar que las aplicaciones nunca utilicen la reconexión en tales casos.

  MQRC_CALL_INTERRUPTED
  Se devuelve este código de razón cuando la conexión se interrumpe durante la ejecución de la llamada Commit y el cliente se vuelve a conectar. Una MQPUT de un mensaje persistente que se encuentre fuera del punto de sincronización también da como resultado que se devuelva a la aplicación el mismo código de razón.

Gestores de colas de alta disponibilidad

Los gestores de colas de alta disponibilidad tienen una instancia activa y una o varias instancias en espera de un gestor de colas. El gestor de colas activo se sincroniza con los gestores de colas en espera, de forma que un gestor de colas en espera puede asumir el control automáticamente si la instancia activa falla. Existe una serie de distintas soluciones para proporcionar gestores de colas de alta disponibilidad, consulte “Configuraciones de alta disponibilidad” en la página 501.

Puede simplificar el reinicio de aplicaciones IBM MQ MQI client, después de que un gestor de colas de alta disponibilidad haya activado su instancia en espera, utilizando la reconexión de cliente automática.

La instancia en espera de un gestor de colas de alta disponibilidad normalmente está en una dirección de red diferente de la instancia activa. Incluya las direcciones de red de ambas instancias en la tabla de definiciones de canal de cliente (CCDT). Proporcione una lista de direcciones de red para el parámetro **CONNAME** o defina varias filas para el gestor de colas en la CCDT. Los gestores de colas de datos replicados y los gestores de colas de alta disponibilidad de IBM MQ Appliance dan soporte a las direcciones IP flotantes, donde se especifica una sola dirección para su uso con gestores de colas activos o en espera.

Grupos de gestores de colas

Normalmente, los IBM MQ MQI clients se reconectan a cualquier gestor de colas de un grupo de gestores de colas. Puede que a veces desee que un IBM MQ MQI client se reconecte únicamente al mismo gestor de colas. Puede tener una afinidad con un gestor de colas.

Puede seleccionar si la aplicación cliente se conecta y reconecta siempre a un gestor de colas del mismo nombre, al mismo gestor de colas, o si se conecta a alguno de un conjunto de gestores de colas que se definen con el mismo valor QMNAME en la tabla de conexiones de clientes.

- El atributo de nombre del gestor de colas, QMNAME, en la definición de canal de cliente es el nombre de un grupo de gestores de colas.
- En la aplicación cliente, si establece el valor del parámetro MQCONN o MQCONNX QmgrName para un nombre de gestores de cola, el cliente se conecta sólo a gestores de colas con ese nombre. Si se prefija el nombre del gestor de colas con un asterisco (*), el cliente se conecta a cualquier grupo de gestores de colas con el mismo valor QMNAME. Para obtener una explicación completa, consulte Grupos de gestores de colas en CCDT.

Puede evitar que un cliente se conecte a un gestor de colas diferente. Establezca la opción MQCNO, MQCNO_RECONNECT_Q_MGR. El IBM MQ MQI client falla si se reconecta a un gestor de colas diferente. Si establece la opción MQCNO, MQCNO_RECONNECT_Q_MGR, no incluya otros gestores de colas en el mismo grupo de gestores de colas. El cliente devuelve un error si el gestor de colas al que se vuelve a conectar no es el mismo gestor de colas al que se conectó en primera instancia.

Grupos de compartición de colas

z/OS La reconexión automática de cliente a los grupos de compartición de colas de z/OS utiliza los mismos mecanismos para la reconexión que cualquier otro entorno. El cliente se volverá a conectar a la misma selección de gestores de colas tal como se haya configurado para la conexión original. Por ejemplo, cuando se utiliza la tabla de definiciones de canal de cliente, el administrador debe asegurarse de que todas las entradas de la tabla se resuelvan en el mismo grupo de compartición de colas de z/OS.

Definiciones de canales de clientes y servidores

Las definiciones de canal de cliente y servidor definen los grupos de gestores de colas a los que puede reconectarse una aplicación cliente. Las definiciones determinan la selección y la temporización de las reconexiones, y otros factores como, por ejemplo, la seguridad; consulte los temas relacionados. Los atributos de canal más relevantes que hay que considerar para la reconexión se enumeran en dos grupos:

Atributos de conexiones de cliente

Afinidad de conexiones (AFFINITY) AFFINITY
Afinidad de conexiones.

Peso de canal de cliente (CLNTWGHT) CLNTWGHT
Peso de canales de cliente.

Nombre de conexión (CONNAME) CONNAME
Información de conexión.

Intervalo de pulsaciones (HBINT) HBINT
Intervalo de pulsaciones. Establezca el intervalo de pulsaciones en el canal de conexión de servidor.

Intervalo de estado activo (KAINT) KAINT
Intervalo de estado activo. Establezca el intervalo de estado activo en el canal de conexión de servidor.

z/OS Tenga en cuenta que KAINT sólo se aplica a z/OS.

Nombre del gestor de colas (QMNAME) QMNAME
Nombre del gestor de colas.

Atributos de conexiones de servidor

Intervalo de pulsaciones (HBINT) HBINT
Intervalo de pulsaciones. Establezca el intervalo de pulsaciones en el canal de conexión de cliente.

Intervalo de estado activo (KAINT) KAINT

Intervalo de estado activo. Establezca el intervalo de estado activo en el canal de conexión de cliente.

z/OS Tenga en cuenta que KAINT sólo se aplica a z/OS.

KAINT es una pulsación de capa de red, y HBINT es una pulsación de IBM MQ entre el cliente y el gestor de colas. Establecer estas pulsaciones en un periodo de tiempo más corto tiene dos objetivos:

1. Al simular la actividad en la conexión, es menos probable que el software de capa de red que busca conexiones no activas que cerrar, cierre su conexión.
2. Si se cierra la conexión, el tiempo antes de que se detecte la conexión interrumpida es menor.

El intervalo keepalive de TCP/IP predeterminado es de dos horas. Considere establecer los atributos KAINT y HBINT en un tiempo más breve. No presuponga que el comportamiento normal de una red se adecua a las necesidades de la reconexión automática. Por ejemplo, algunos cortafuegos pueden cerrar una conexión TCP/IP no activa después de sólo diez minutos.

Conectividad de red

Sólo las anomalías de red que la red pasa al IBM MQ MQI client son manejadas por la funcionalidad de reconexión automática del cliente.

- Las reconexiones que el transporte realiza de forma automática son invisibles para IBM MQ.
- Establecer HBINT ayuda a resolver anomalías de red que son invisibles para IBM MQ.

Gestores de colas y escuchas de IBM MQ

La reconexión de cliente la desencadena una anomalía de servidor, una anomalía de gestor de colas, una anomalía de conectividad de red y un administrador que cambie a otra instancia de gestor de colas.

- Si está utilizando un gestor de colas multiinstancia, existe otra causa por la que puede ocurrir la reconexión de cliente cuando se cambia el control de una instancia de gestor de colas activo a una instancia en espera.
- La finalización de un gestor de colas mediante el mandato **endmqm** predeterminado, no desencadena la reconexión automática del cliente. Añada la opción `-r` en el mandato **endmqm** para solicitar la reconexión automática de cliente, o la opción `-s` para transferir a una instancia de gestor de colas de reserva tras la conclusión.

Soporte de reconexión automática de IBM MQ MQI client

Si utiliza el soporte de reconexión automática de cliente en el IBM MQ MQI client, la aplicación cliente se reconecta automáticamente y continúa el proceso sin necesidad de emitir una llamada MQI MQCONN o MQCONNX para reconectarse al gestor de colas.

- La reconexión automática de cliente se desencadena debido a una de las causas siguientes:
  - Error del gestor de colas
  - Finalizar un gestor de colas y especificar la opción `-r`, volver a conectar, en el mandato **endmqm**.
- Las opciones MQCNO de MQCONNX controlan si se ha habilitado la reconexión automática de cliente. Las opciones se describen en Opciones de reconexión.
- La reconexión automática de cliente emite llamadas MQI en nombre de la aplicación para restaurar el manejador de conexiones y los manejadores de otros objetos abiertos, para que el programa pueda reanudar el proceso normal después de procesar cualquier error MQI que haya podido resultar tras la interrupción de la conexión. Consulte “Recuperación de un cliente reconectado automáticamente” en la página 590.
- Si ha escrito un programa de salida de canal para la conexión, la salida recibe estas llamadas MQI adicionales.
- Puede registrar un manejador de sucesos de reconexión, que se desencadena cuando la reconexión comienza y cuando finaliza.

Aunque el tiempo de reconexión previsto es poco más de un minuto, la reconexión puede tardar más tiempo, ya que un gestor de colas puede tener que gestionar numerosos recursos. Durante este tiempo, una aplicación cliente podría estar manteniendo bloqueos que no pertenezcan a recursos de IBM MQ. Existe un valor de tiempo de espera que puede configurar para limitar el tiempo que un cliente espera la reconexión. El valor (en segundos) se establece en el archivo `mqclient.ini`.

```
Channels:
MQReconnectTimeout = 1800
```

Una vez que el tiempo de espera ha finalizado, no hay más intentos de reconexión. Cuando el sistema detecta que el tiempo de espera ha finalizado, devuelve un error `MQRC_RECONNECT_FAILED`.

Conceptos relacionados

Clientes que se pueden volver a conectar

Tareas relacionadas

Detención de un gestor de colas

z/OS Console message monitoring

On IBM MQ for z/OS, there are a number of information messages issued by the queue manager or channel initiator that should be considered particularly significant. These messages do not in themselves indicate a problem, but can be useful in tracking because they do indicate a potential issue which might need addressing.

The presence of these console messages might also indicate that a user application is putting a large number of messages to the page set, which might be a symptom of a larger problem:

- A problem with the user application which PUTs messages, such as an uncontrolled loop.
- A user application which GETs the messages from the queue is no longer functioning.

Console messages to monitor

The following list outlines messages which can potentially indicate larger problems. Determine if it is necessary to track these messages with system automation and provide appropriate documentation so any potential problems can be followed up effectively.

CSQI004I: *csect-name* CONSIDER INDEXING *queue-name* BY *index-type* FOR *connection-type* CONNECTION *connection-name*, *num-msgs* MESSAGES SKIPPED

- The queue manager has detected an application receiving messages by message ID or correlation ID from a queue that does not have an index defined.
- Consider establishing an index for the identified queue by altering the local queue object, *queue-name*, INDXTYPE attribute to have value *index-type*.

CSQI031I: *csect-name* THE NEW EXTENT OF PAGE SET *psid* HAS FORMATTED SUCCESSFULLY

- Check the curdepth of the queues allocated to this page set.
- Investigate the cause of the failure to process the messages.

.

CSQI041I: *csect-name* JOB *jobname* USER *userid* HAD ERROR ACCESSING PAGE SET *psid*

- Determine if the page set is allocated to the queue manager.
- Issue a **DISPLAY USAGE** command to determine the state of the page set.
- Check the queue manager joblog for additional error messages.

.

CSQI045I: *csect-name* Log RBA has reached *rba*. Plan a log reset

- Plan to stop the queue manager at a convenient time and reset the logs.

- If your queue manager is using 6-byte log RBAs, consider converting the queue manager to use 8-byte log RBAs.

CSQI046E: *csect-name* Log RBA has reached *rba*. Perform a log reset

- Plan to stop the queue manager at a convenient time and reset the logs.
- If your queue manager is using 6-byte log RBAs, consider converting the queue manager to use 8-byte log RBAs.

CSQI047E: *csect-name* Log RBA has reached *rba*. Stop queue manager and reset logs

- Stop the queue manager immediately and reset the logs.
- If your queue manager is using 6-byte log RBAs, consider converting the queue manager to use 8-byte log RBAs.

CSQJ004I: ACTIVE LOG COPY *n* INACTIVE, LOG IN SINGLE MODE, ENDRBA= *ttt*

- The queue manager has activated 'single' logging mode. This is often indicative of a log offload problem.
- Issue a **DISPLAY LOG** command to determine your settings for duplexing of active and archive logs. This display also shows how many active logs need offload processing.
- Check the queue manager joblog for additional error messages

CSQJ031D: *csect-name*, THE LOG RBA RANGE MUST BE RESET. REPLY 'Y' TO CONTINUE STARTUP OR 'N' TO SHUTDOWN

- Stop the queue manager and reset the logs as soon as possible and reset the logs.
- If your queue manager is using 6-byte log RBAs, consider converting the queue manager to use 8-byte log RBAs.

CSQJ032E: *csect-name alert-lvl* - APPROACHING END OF THE LOG RBA RANGE OF *max-rba*. CURRENT LOG RBA IS *current-rba*.

- Plan to stop the queue manager and reset the logs as soon as possible.
- If your queue manager is using 6-byte log RBAs, consider converting the queue manager to use 8-byte log RBAs.

CSQJ110E: LAST COPY*n* ACTIVE LOG DATA SET IS *nnn* PERCENT FULL

- Take steps to complete other waiting offload tasks by performing a display request to determine the outstanding requests related to the log offload process. Take the necessary action to satisfy any requests, and permit offload to continue.
- Consider whether there are sufficient active log data sets. If necessary, you can add additional log data sets dynamically by using the DEFINE LOG command.

CSQJ111A: OUT OF SPACE IN ACTIVE LOG DATA SETS

- Perform a display request to ensure that there are no outstanding requests that are related to the log offload process. Take the necessary action to satisfy any requests, and permit offload to continue.
- Consider whether there are sufficient active log data sets. If necessary, you can add additional log data sets dynamically by using the DEFINE LOG command.
- If the delay was caused by the lack of a resource required for offload, the necessary resource must be made available to allow offload to complete and thus permit logging to proceed. For information about recovery from this condition, see Archive log problems.

CSQJ114I: ERROR ON ARCHIVE DATA SET, OFFLOAD CONTINUING WITH ONLY ONE ARCHIVE DATA SET BEING GENERATED

- Check the queue manager joblog for additional error messages.
- Make a second copy of the archive log and update your BSDS manually.

CSQJ115E: OFFLOAD FAILED, COULD NOT ALLOCATE AN ARCHIVE DATA SET

Review the error status information of message CSQJ103E or CSQJ073E. Correct the condition that caused the data set allocation error so that, on retry, the offload can take place.

CSQJ136I: UNABLE TO ALLOCATE TAPE UNIT FOR CONNECTION-ID= *xxxx* CORRELATION-ID= *yyyyyy*, *m* ALLOCATED *n* ALLOWED

- Check the queue manager joblog for additional error messages.

.

CSQJ151I: *csect-name* ERROR READING RBA *rrr*, CONNECTION-ID= *xxxx* CORRELATION-ID= *yyyyyy* REASON CODE= *ccc*

- Check the queue manager joblog for additional messages.
- Issue a **DISPLAY CONN** command to determine which connection is not committing its activity.
- Ensure the application can commit its updates.

CSQJ160I: LONG-RUNNING UOW FOUND, URID= *urid* CONNECTION NAME= *name*

- Check the queue manager joblog for additional messages.
- Issue a **DISPLAY CONN** command to determine which connection is not committing its activity.
- Ensure the application can commit its updates.

CSQJ161I: UOW UNRESOLVED AFTER *n* OFFLOADS, URID= *urid* CONNECTION NAME= *name*

- Determine if the page set is allocated to the queue manager.
- Issue a **DISPLAY USAGE** command to determine the state of the page set.
- Check the queue manager joblog for additional messages.

CSQP011E: CONNECT ERROR STATUS *ret-code* FOR PAGE SET *psid*

- Check the curdepth of the queues allocated to this page set.
- Investigate the cause of the failure to process messages.

CSQP013I: *csect-name* NEW EXTENT CREATED FOR PAGE SET psid. NEW EXTENT WILL NOW BE FORMATTED

- Check the curdepth of the queues allocated to this page set.
- Investigate the cause of failure to process messages.
- Determine if queues need to be relocated to another page set.
- If the volume is full, determine if you need to make the page set a multi volume data set. If the page set is already multi-volume, consider adding more volumes to the storage group being used. Once more space is available retry the expansion by setting the page set **EXPAND** method to **SYSTEM**. If a retry is required, toggle **EXPAND** to **SYSTEM** and then back to your normal setting.

CSQP014E: *csect-name* EXPANSION FAILED FOR PAGE SET psid. FUTURE REQUESTS TO EXTEND IT WILL BE REJECTED

- Check the curdepth of the queues allocated to this page set.
- Investigate the cause of failure to process messages.
- Determine if queues need to be relocated to another page set.

CSQP016E: *csect-name* PAGE SET *psid* HAS REACHED THE MAXIMUM NUMBER OF EXTENTS. IT CANNOT BE EXTENDED AGAIN

- Check the curdepth of the queues allocated to this page set.
- Investigate the cause of failure to process messages.

CSQP017I: *csect-name* EXPANSION STARTED FOR PAGE SET *psid*

Issue DISPLAY THREAD commands to determine the state of the Units of Work in IBM MQ.

CSQP047E: Unavailable page sets can cause problems - take action to correct this situation

- Follow the system programmer response.

CSQQ008I: *nn* units of recovery are still in doubt in queue manager *qqqq*

- Investigate the state of your dead letter queue. Ensure the dead letter queue is not PUT disabled.
- Ensure the dead letter queue is not at the MAXMSG limit.

CSQQ113I: *psb-name region-id* This message cannot be processed

- Check the CSQOUTX data set to determine the cause of the CSQINPX failure.
- Some commands may not be processed.

CSQX035I: *csect-name* Connection to queue manager *qmgr-name* stopping or broken, MQCC= *mqcc* MQRC= *mqrc* (*mqrc-text*)

- Check the MQRC to determine the cause of the failure.
- These codes are documented in IBM MQ for z/OS messages, completion, and reason codes.

CSQX032I: *csect-name* Initialization command handler terminated

- Check the MQRC to determine the cause of the failure.
- These codes are documented in IBM MQ for z/OS messages, completion, and reason codes.

CSQX048I: *csect-name* Unable to convert message for *name*, MQCC= *mqcc* MQRC= *mqrc* (*mqrc-text*)

- Check the joblog to determine the cause of the TCP/IP failure.
- Check the TCP/IP address space for errors.

CSQX234I: *csect-name* Listener stopped, TRPTYPE= *trptype* INDISP= *disposition*

- If the listener does not stop, following a **STOP** command, check the TCP/IP address space for errors.
- Follow the system programmer response.

CSQX407I: *csect-name* Cluster queue *q-name* definitions inconsistent

- Multiple cluster queues within the cluster have inconsistent values. Investigate and resolve the differences.

CSQX411I: *csect-name* Repository manager stopped

- If the repository manager has stopped because of an error, check the joblog for messages.

CSQX417I: *csect-name* Cluster-senders remain for removed queue manager *qmgr-name*

- Follow the system programmer response.

CSQX418I: *csect-name* Only one repository for cluster *cluster_name*

- For increased high availability, clusters should be configured with two full repositories.

CSQX419I: *csect-name* No cluster-receivers for cluster *cluster_name*

- Follow the system programmer response.

CSQX420I: *csect-name* No repositories for cluster *cluster_name*

- Follow the system programmer response.

CSQX448E: *csect-name* Repository manager stopping because of errors. Restart in *n* seconds

- Follow the system programmer response.

  This message is put out every 600 seconds (10 minutes) until the SYSTEM.CLUSTER.COMMAND.QUEUE is enabled, by using the command:

```
ALTER QLOCAL(SYSTEM.CLUSTER.COMMAND.QUEUE) GET(ENABLED)
```

Before enabling the queue, manual intervention might be required to resolve the problem that caused the repository manager to end, prior to the first CSQX448E message being issued.

CSQX548E: *csect-name* Messages sent to local dead-letter queue, channel *channel-name* reason=*mqrc* (*mqrc-text*)

- Follow the system programmer response.

CSQX788I: *csect-name* DNS lookup for address *address* using function '*func*' took *n* seconds

- Follow the system programmer response.

CSQY225E: *csect-name* Queue manager is critically short of local storage above the bar - take action

- The queue manager is running critically short of virtual storage above the bar. Action should be taken to relieve the situation, and to avoid the possible abnormal termination of the queue manager.

CSQ5038I: *csect-name* Service task service-task has been unresponsive since hh.mm.ss.nnnnnn. Check for problems with Db2

- Follow the system programmer response.

Configuraciones de alta disponibilidad

Si desea utilizar sus gestores de colas de IBM MQ en una configuración de alta disponibilidad (HA), puede configurar sus gestores de colas para que funcionen con un gestor de alta disponibilidad, como PowerHA para AIX (antes HACMP) o Microsoft Cluster Service (MSCS), o con gestores de colas multiinstancia de IBM MQ. En sistemas Linux, también se pueden desplegar los gestores de colas de datos replicados (RDQM), que utilizan un grupo basado en quórum para proporcionar una alta disponibilidad. Otra opción, HA nativa, está dirigida a los despliegues de contenedor.

MQ Appliance Otra opción para una solución de alta disponibilidad o recuperación tras desastre es desplegar un par de dispositivos IBM MQ. Consulte Alta disponibilidad y Recuperación tras desastre en la documentación de IBM MQ Appliance.

Debe tener en cuenta la siguientes definiciones de configuración:

Clústeres del gestor de colas
Grupos de dos o más gestores de colas en uno o varios sistemas que proporcionan una interconexión automática y que permiten que se compartan las colas entre ellos con fines de equilibrio de carga y redundancia. A partir de IBM WebSphere MQ 7.1, la recuperación de errores de clúster vuelve a ejecutar las operaciones que han causado problemas hasta que estos se resuelven.

Clústeres HA
Los clústeres HA son grupos de dos o más sistemas y recursos, como discos y redes, conectados entre sí y configurados de forma que si uno falla, un gestor de alta disponibilidad, como HACMP (AIX and Linux) o MSCS (Windows) realiza una *sustitución por anomalía*. La sustitución por anomalía transfiere los datos de estado de las aplicaciones del sistema anómalo a otro sistema del clúster y reinicia allí la operación. Esto proporciona una alta disponibilidad de servicios que se ejecutan en el clúster HA. La relación entre los clústeres de IBM MQ y los clústeres HA se describe en "Relación entre los clústeres HA y los clústeres de gestores de colas" en la página 503.

Gestores de colas multiinstancia
Instancias del mismo gestor configurado en dos o más sistemas. Al iniciar varias instancias, una se convierte en la instancia activa y las otras se convierten en instancias en espera. Si la instancia activa falla, la sustituye automáticamente una instancia en espera que se esté ejecutando en un sistema diferente. Puede utilizar gestores de colas multiinstancia para configurar sus propios sistemas de mensajería de alta disponibilidad basados en IBM MQ, sin necesidad de una tecnología de clúster, como HACMP o MSCS. Los clústeres HA y los gestores de colas multiinstancia son formas alternativas de que los gestores de colas tengan una alta disponibilidad. No los combine colocando un gestor de colas multiinstancia en un clúster HA.

Gestores de colas de datos replicados de alta disponibilidad (RDQM de HA)
Instancias del mismo gestor configurado en cada nodo de un grupo de tres servidores Linux. Una de las tres instancias es la instancia activa. Los datos del gestor de colas activo se replican de forma

síncrona hacia las otras dos instancias, para que una de dichas instancias pueda tomar el relevo en caso de producirse algún fallo. La agrupación de los servidores está controlada por Pacemaker; la réplica, por DRBD.

Gestores de colas de datos replicados de recuperación tras desastre (RDQM de DR)
Un gestor de colas se ejecuta en un nodo primario en un sitio, con una instancia secundaria de ese gestor de colas ubicada en un nodo de recuperación en otro sitio. Los datos se replican entre la instancia primaria y la instancia secundaria, y si el nodo primario se pierde por cualquier motivo, la instancia secundaria puede convertirse en la instancia primaria e iniciarse. Ambos nodos deben ser servidores Linux. La réplica la controla DRBD.

Gestores de colas de datos replicados de recuperación tras desastre/alta disponibilidad (RDQM de DR/HA)
Puede configurar un gestor de colas de datos replicados (RDQM) que se ejecute en un grupo de alta disponibilidad en un sitio, pero que pueda migrar tras error a otro grupo de alta disponibilidad en otro sitio si se produce algún desastre que hace que el primer grupo no esté disponible. Esto se conoce como RDQM de DR/HA.

CP4I **HA nativa**
HA nativa es una solución de alta disponibilidad destinada a los despliegues de contenedor de IBM MQ. HA nativa utiliza la réplica de registro para mantener actualizadas tres instancias de un gestor de colas que se ejecutan en nodos distintos. Una instancia está activa en cualquier momento y procesa los mensajes. El gestor de colas activo envía sus actualizaciones de registro a las otras dos instancias para mantenerlas actualizadas. Si la instancia activa falla, una de las instancias de réplica asume automáticamente el rol activo.

Diferencias entre gestores de colas multiinstancia y clústeres HA

Los gestores de colas multiinstancia y los clústeres HA son formas alternativas de conseguir una alta disponibilidad para los gestores de colas. A continuación, enumeramos algunos puntos que subrayan las diferencias principales entre los dos sistemas.

Los gestores de colas multiinstancia incluyen las características siguientes:

- Soporte de migración tras error básico integrado en IBM MQ
- Una sustitución por anomalía más rápida que el clúster HA
- Una configuración y un funcionamiento simples
- Integración con IBM MQ Explorer

Algunas de las limitaciones de los gestores de colas multiinstancia son las siguientes:

- Se precisa de un almacenamiento en red de alta disponibilidad y alto rendimiento
- La configuración de red es más compleja porque el gestor de colas cambia de dirección IP cuando realiza una sustitución por anomalía

Los clústeres HA incluyen las características siguientes:

- La posibilidad de coordinar varios recursos, como, por ejemplo, un servidor de aplicaciones o una base de datos
- Unas opciones de configuración más flexibles incluidos los clústeres que constan de más de dos nodos
- Puede sustituirse por anomalía varias veces sin necesidad de que intervenga el operador
- Toma de control de la dirección IP del gestor de colas como parte de la sustitución por anomalía

Las limitaciones de los clústeres HA son las siguientes:

- Se necesitan conocimientos y compra de productos adicionales
- Se precisan discos que se puedan intercambiar entre los nodos del clúster
- La configuración de los clústeres HA es bastante compleja
- La sustitución por anomalía es bastante lenta históricamente, pero los recientes productos de clúster HA están mejorando

- Se pueden producir sustituciones por anomalía innecesarias si se producen fallos en los scripts que se utilizan para supervisar recursos, como por ejemplo, gestores de colas

Relación entre los clústeres HA y los clústeres de gestores de colas

Los clústeres de gestores de colas ofrecen un equilibrio de carga de los mensajes en todas las instancias de colas de clústeres de gestores de colas. Esto permite una disponibilidad más alta que con un único gestor de colas ya que, tras una anomalía de un gestor de colas, las aplicaciones de mensajería todavía pueden acceder a las instancias que quedan de una cola de clústeres de gestores de colas. Sin embargo, aunque los clústeres de gestores de colas direccionan automáticamente los nuevos mensajes a los gestores de colas disponibles de un clúster, los mensajes que actualmente están en cola en un gestor de colas no disponible no estarán disponibles hasta que se reinicie dicho gestor de colas. Por este motivo, los clústeres de gestores de colas por sí solos no proporcionan alta disponibilidad de todos los datos de mensajes ni proporcionan la detección automática de errores del gestor de colas ni el desencadenamiento del reinicio o de la sustitución por anomalía del gestor de colas. Los clústeres de alta disponibilidad(HA) proporcionan estas características. Los dos tipos de clústeres se pueden utilizar conjuntamente con un buen resultado. Para obtener una introducción a los clústeres de gestores de colas, consulte Diseño de clústeres.

Conceptos relacionados

Alta disponibilidad para IBM MQ Advanced container

Clústeres HA en AIX and Linux

Puede utilizar IBM MQ con un clúster de alta disponibilidad (HA) en plataformas AIX and Linux: por ejemplo, PowerHA para AIX (antes conocido como HACMP), Veritas Cluster Server, HP Serviceguard o un clúster Red Hat Enterprise Linux con Red Hat Cluster Suite.

En esta sección se introducen los temas siguientes: “Configuraciones de clústeres HA” en la página 503, la relación entre los clústeres HA y los clústeres de gestores de colas, “Clientes de IBM MQ” en la página 504 y “Funcionamiento de IBM MQ en un clúster HA” en la página 504, se recorren los pasos de configuración necesarios y se ofrecen scripts de ejemplo que puede adaptar para configurar gestores de colas con un clúster HA.

Para obtener ayuda sobre los pasos de configuración que se describen en esta sección, consulte la documentación del clúster HA correspondiente a su entorno.

Configuraciones de clústeres HA

En esta sección, el término *nodo* se utiliza para hacer referencia a la entidad que ejecuta un sistema operativo y el software de HA; "PC", "sistema", "máquina", "partición" o "Blade" pueden considerarse que son sinónimos. Puede utilizar IBM MQ para ayudar a establecer configuraciones en espera o de toma de control, incluida la toma de control mutua, en la que todos los nodos de clúster ejecutan carga de trabajo de IBM MQ.

Una configuración *en espera* es la configuración de clúster HA más básica en la que un nodo realiza el trabajo mientras que el otro está en espera. El nodo en espera no desempeña ningún trabajo y recibe el nombre de nodo desocupado; esta configuración a veces se llama *espera en frío*. Una configuración de este tipo requiere un alto grado de redundancia de hardware. Para economizar hardware, es posible ampliar esta configuración de modo que haya varios nodos de trabajo con un solo nodo en espera. La única razón es que el nodo en espera puede retomar el trabajo de cualquier otro nodo de trabajo. Esta configuración sigue llamándose configuración en espera y a veces se denomina configuración "N+1".

Una configuración *de toma de control* es una configuración más avanzada en la que todos los nodos realizan algún trabajo y se retoma un trabajo importante en caso de una anomalía de nodo.

Una configuración de *toma de control unilateral* es aquella en la que un nodo en espera realiza algún trabajo adicional, no crucial e inamovible. Esta configuración es similar a una configuración en espera, salvo que el trabajo (no crucial) lo desempeña el nodo en espera.

Una configuración de *toma de control mutua* es aquella en la que todos los nodos desempeñan un trabajo de alta disponibilidad (movible). Este tipo de configuración de clúster HA también recibe el nombre de "Activa/Activa" para indicar que todos los nodos procesan de forma activa una carga de trabajo crucial.

Con la configuración en espera ampliada o cualquiera de las dos configuraciones de toma de control es importante tener en cuenta la carga máxima que se puede asignar a un nodo que puede retomar el trabajo de otros nodos. Este nodo debe tener suficiente capacidad para mantener un nivel razonable de rendimiento.

Relación entre los clústeres HA y los clústeres de gestores de colas

Los clústeres de gestores de colas reducen las tareas de administración y ofrecen un equilibrio de carga de los mensajes a través de instancias de colas de clúster de gestores de colas. También ofrecen una mayor disponibilidad que un único gestor de colas porque, tras una anomalía de un gestor de colas, las aplicaciones de mensajería todavía pueden acceder a las instancias que quedan de una cola de clúster de gestores de colas. Pero los clústeres de gestores de colas por sí solos no permiten detectar automáticamente el error de los gestores de colas y la activación automática del reinicio o la sustitución por anomalía del gestor de colas. Los clústeres HA proporcionan estas características. Los dos tipos de clústeres se pueden utilizar conjuntamente con un buen resultado.

Clientes de IBM MQ

Los clientes de IBM MQ que se comuniquen con un gestor de colas que pudiera ser objeto de un reinicio o una toma de control deben estar escritos para tolerar una conexión interrumpida y deben intentar repetidamente reconectarse. IBM MQincluye características en el proceso de la tabla de definiciones de canal de cliente (CCDT) que ayudan con la disponibilidad de la conexión y el equilibrio de carga de trabajo; sin embargo, estas no son directamente relevantes cuando se trabaja con un sistema de migración tras error.

La funcionalidad transaccional permite que un IBM MQ MQI client participe en transiciones de dos fases, siempre que el cliente esté conectado con el mismo gestor de colas. La funcionalidad transaccional no puede utilizar técnicas, como un equilibrador de carga IP, para seleccionar entre una lista de gestores de colas. Cuando se utiliza un producto HA, un gestor de colas mantiene su identidad (nombre e identidad), sea cual sea el nodo en el que se ejecute, por lo tanto la funcionalidad transaccional se puede utilizar con gestores de colas que están bajo control de HA.

Funcionamiento de IBM MQ en un clúster HA

Todos los clústeres HA tienen el concepto de unidad de sustitución por anomalía. Se trata de un conjunto de definiciones que contiene todos los recursos que forman el servicio de alta disponibilidad. La unidad de sustitución por anomalía incluye el propio servicio y todos los demás recursos de los que depende.

Las soluciones HA utilizan diferentes términos para una unidad de sustitución por anomalía:

- En PowerHA para AIX, la unidad de región propietaria del archivo recibe el nombre de *grupo de recursos*.
- En Veritas Cluster Server se conoce como *grupo de servicios*.
- En Serviceguard se denomina *paquete*.

Este tema utiliza el término *grupo de recursos* para hacer referencia a una unidad de sustitución por anomalía.

La unidad más pequeña de sustitución por anomalía para IBM MQ es un gestor de colas. Normalmente, el grupo de recursos que contiene el gestor de colas también contiene discos compartidos en un grupo de volúmenes o un grupo de discos que está reservado exclusivamente para el uso del grupo de recursos, y la dirección IP que se utiliza para conectar al gestor de colas. También es posible incluir otros recursos de IBM MQ, como por ejemplo un supervisor desencadenante o un escucha en el mismo grupo de recursos, ya sea como recursos separados o bajo el control del propio gestor de colas.

Los datos y registros de un gestor de colas que se va a utilizar en un clúster HA deben estar en discos que se comparten entre los nodos del clúster. El clúster HA se encarga de que sólo un nodo del clúster a la vez pueda grabar en los discos. El clúster HA puede utilizar un script de supervisor para supervisar el estado del gestor de colas.

Es posible utilizar un único disco compartido para los datos y los registros que están relacionados con el gestor de colas. Pero lo habitual es utilizar sistemas de archivos compartidos separados para que se pueda calcular su tamaño y se puedan ajustar de forma independiente.

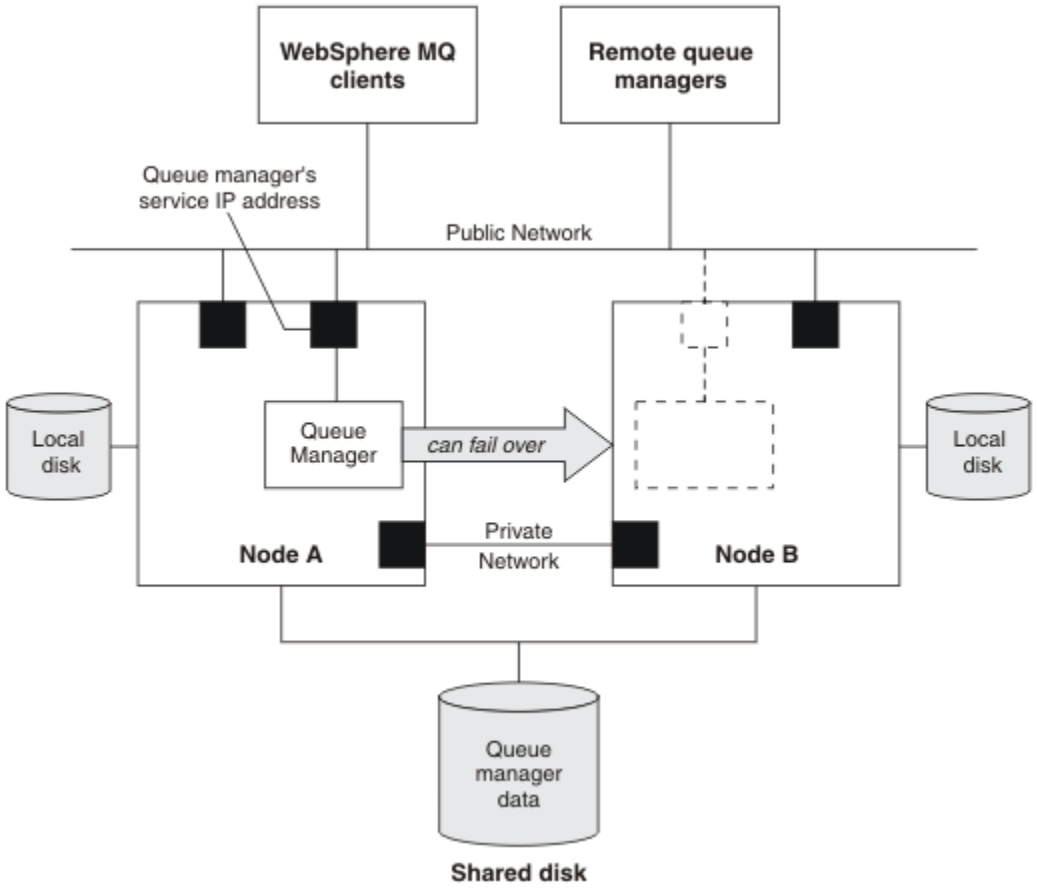


Figura 69. Clúster HA

La figura 1 muestra un clúster HA con dos nodos. El clúster HA gestiona la disponibilidad de un gestor de colas que se ha definido en un grupo de recursos. Es una configuración en espera activa/pasiva o en frío porque sólo hay un nodo, el nodo A que ejecuta actualmente un gestor de colas. El gestor de colas se creó con datos y archivos de registro en un disco compartido. El gestor de colas tiene una dirección IP de servicio que también está gestionada por el clúster HA. El gestor de colas depende del disco compartido y de la dirección IP de servicio. Cuando el clúster HA sustituye al gestor de colas desde el nodo A hasta el nodo B, primero mueve los recursos dependientes del gestor de colas al nodo B y, a continuación, inicia el gestor de colas.

Si el clúster HA contiene más de un gestor de colas, la configuración del clúster HA puede hacer que dos o más gestores de colas se ejecuten en el mismo nodo tras una anomalía. A cada gestor de colas del clúster HA debe asignarse a su propio número de puerto, que utiliza en cualquier nodo de clúster que esté activo en cualquier momento específico.

Normalmente, el clúster HA se ejecuta como el usuario root. IBM MQ se ejecuta como el usuario mqm. La administración de IBM MQ se otorga a miembros del grupo mqm. Asegúrese de que el usuario y el grupo mqm existen ambos en todos los nodos de clúster HA. El ID de usuario y el ID de grupo deben ser coherentes en el clúster. La administración de IBM MQ por parte del usuario root no está permitida; los scripts que inician, detienen o supervisan scripts deben cambiar al usuario mqm.

Nota: IBM MQ debe instalarse correctamente en todos los nodos; no se pueden compartir los archivos ejecutables del producto.

Linux AIX *Configuración de discos compartidos en AIX and Linux*

Un gestor de colas IBM MQ en un clúster de alta disponibilidad (HA) requiere que los archivos de datos y los archivos de registro residan en sistemas de archivos remotos con nombres comunes en un disco compartido.

Acerca de esta tarea

la Figura 1 muestra un posible diseño de un gestor de colas en un clúster HA. Los datos y los directorios de registros del gestor colas están los dos en el disco compartido que se monta en /MQHA/QM1. Este disco se intercambia entre los nodos del clúster HA cuando se produce la sustitución por anomalía con lo cual los datos están disponibles cada vez que se reinicia el gestor de colas. El archivo `mqs.ini` tiene una stanza para el gestor de colas de QM1. La stanza Log del archivo `qm.ini` tiene un valor para LogPath.

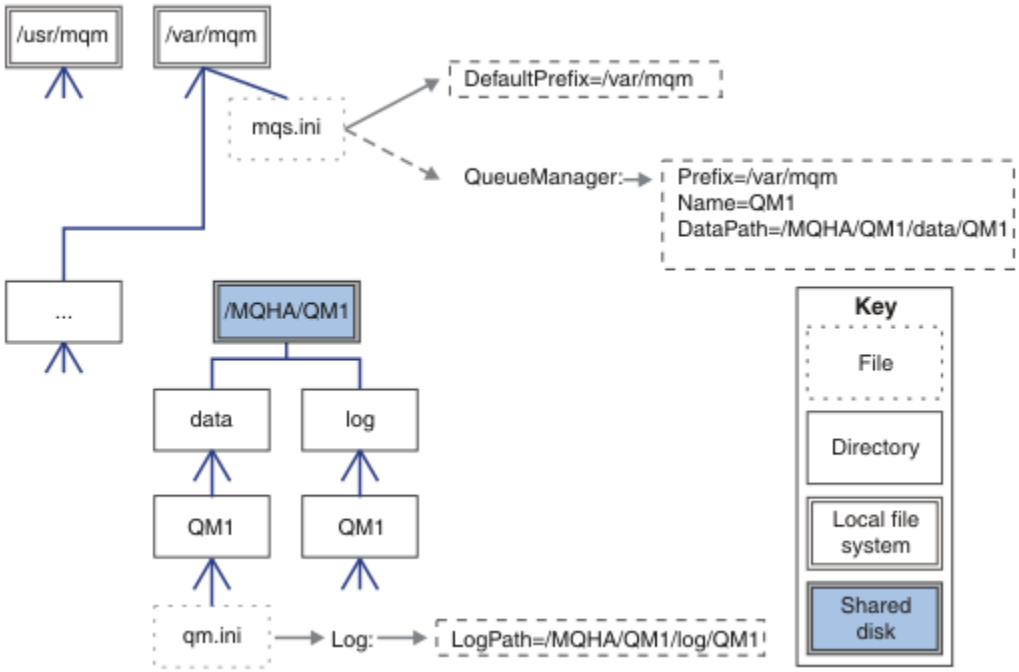


Figura 70. Directorios compartidos con nombre `data` y `log`

Procedimiento

1. Decida los nombres de los puntos de montaje de los sistemas de archivos del gestor de colas.

   Por ejemplo, `/MQHA/qmgrname/data` para los archivos de datos del gestor de colas y `/MQHA/qmgrname/log` para sus archivos de registro.
2. Cree un grupo de volúmenes (o grupo de discos) que contenga los datos y los archivos de registro del gestor de colas.

   Este grupo de volúmenes está gestionado por un clúster de alta disponibilidad (HA) en el mismo grupo de recursos que el gestor de colas.
3. Cree los sistemas de archivos para los datos y los archivos de registro del gestor de colas en el grupo de volúmenes.
4. Para cada uno de los nodos, cree los puntos de montaje de los sistemas de archivos y asegúrese de que los sistemas de archivos se pueden montar.

   El usuario mqm debe ser el propietario de los puntos de montaje.

Linux AIX *Creación de un gestor de colas de clúster HA en AIX and Linux*

El primer paso para utilizar un gestor de colas en un clúster de alta disponibilidad es crear el gestor de colas en uno de los nodos.

Acerca de esta tarea

Para crear un gestor de colas para utilizarlo en un clúster HA, primero debe seleccionar uno de los nodos del clúster en el que se va a crear el gestor de colas y, a continuación, complete los pasos siguientes en este nodo.

Procedimiento

1. Monte el sistema de archivos del gestor de colas en el nodo.
2. Cree el gestor de colas mediante el mandato **crtmqm**.

   Por ejemplo:

   ```
   crtmqm -md /MQHA/qmgrname/data -ld /MQHA/qmgrname/log qmgrname
   ```
3. Inicie manualmente el gestor de colas mediante el mandato **strmqm**.
4. Complete cualquier configuración inicial del gestor de colas, como, por ejemplo, crear colas y canales y establecer el gestor de colas para iniciar un escucha automáticamente cuando se inicia el gestor de colas.
5. Detenga el gestor de colas mediante el mandato **endmqm**.
6. Utilice el mandato **dspmqinf** para visualizar el mandato **addmqinf**:

   ```
   dspmqinf -o command qmgrname
   ```

   donde qmgrname es el nombre del gestor de colas.

   Para obtener más información sobre cómo utilizar el mandato **addmqinf** , consulte “Adición de la configuración de gestor de colas a otros nodos de clúster HA en AIX and Linux” en la página 507.

   El mandato **addmqinf** se visualiza de un modo similar al ejemplo siguiente:

   ```
   addmqinf -sQueueManager -vName=qmgrname -vDirectory=qmgrname \
   -vPrefix=/var/mqm -vDataPath=/MQHA/qmgrname/data/qmgrname
   ```
7. Tome buena nota del mandato visualizado.
8. Desmonte los sistemas de archivos del gestor de colas.

Qué hacer a continuación

Ya está preparado para completar los pasos que se describen en el apartado “Adición de la configuración de gestor de colas a otros nodos de clúster HA en AIX and Linux” en la página 507.

Linux AIX *Adición de la configuración de gestor de colas a otros nodos de clúster HA en AIX and Linux*

Debe añadir la información de configuración del gestor de colas a los otros nodos del clúster HA.

Antes de empezar

Para completar esta tarea, debe haber ejecutado los pasos que se describen en “Creación de un gestor de colas de clúster HA en AIX and Linux” en la página 507. Después de haber creado el gestor de colas, debe añadir la información de configuración para el gestor de colas a cada uno de los otros nodos del clúster HA completando los pasos siguientes en cada uno de los demás nodos.

Acerca de esta tarea

Cuando se crea un gestor de colas para utilizarlo en un clúster HA, primero debe seleccionar uno de los nodos del clúster en el que se debe crear el gestor de colas, tal como se describe en "Creación de un gestor de colas de clúster HA en AIX and Linux" en la página 507.

Procedimiento

1. Monte los sistemas de archivos del gestor de colas.
2. Añada la información de configuración del gestor de colas al nodo.

   Hay dos formas de añadir la información de configuración:

   - Editando `/var/mqm/mqs.ini` directamente.
   - Emitiendo el mandato **addmqinf** visualizado por el mandato **dspmqinf** en el paso 6 en "Creación de un gestor de colas de clúster HA en AIX and Linux" en la página 507.
3. Inicie y detenga el gestor de colas para verificar la configuración.

   Los mandatos utilizados para iniciar y detener las instancias del gestor de colas deben emitirse desde la misma instalación de IBM MQ que el mandato **addmqinf**. Para iniciar y detener el gestor colas de una instalación diferente de la que actualmente está asociada con el gestor de colas, primero debe establecer la instalación asociada con el gestor de colas mediante el mandato **setmqm**. Para obtener más información, consulte setmqm.
4. Desmonte los sistemas de archivos del gestor de colas.

Linux AIX *Scripts de shell de ejemplo para iniciar un gestor de colas de clúster HA en AIX and Linux*

El gestor de colas se representa en el clúster HA como un recurso. El clúster HA debe ser capaz de iniciar y detener el gestor de colas. En la mayoría de los casos puede utilizar un script de shell para iniciar el gestor de colas. Estos scripts deben estar disponibles en la misma ubicación en todos los nodos del clúster, utilizando un sistema de archivos de red o copiándolos en cada uno de los discos locales.

Nota: Antes de reiniciar un gestor de colas que haya fallado, debe desconectar las aplicaciones de dicha instancia del gestor de colas. Si no lo hace, es posible que el gestor de colas no se reinicie correctamente.

Aquí se proporcionan ejemplos de scripts de shell adecuados. Puede adaptarlos según sus necesidades y utilizarlos para iniciar el gestor de colas bajo el control del clúster HA.

El siguiente script es un ejemplo de cómo pasar del usuario de clúster HA al usuario mqm para que el gestor de colas pueda iniciarse satisfactoriamente:

```
#!/bin/ksh

# A simple wrapper script to switch to the mqm user.

su mqm -c name_of_your_script $*
```

El siguiente script de shell es un ejemplo de cómo iniciar un gestor de colas sin realizar ninguna presuposición sobre el estado actual del gestor de colas. Observe que utiliza un método extremadamente brusco de finalizar los procesos que pertenecen al gestor de colas:

```
#!/bin/ksh
#
# This script robustly starts the queue manager.
#
# The script must be run by the mqm user.

# The only argument is the queue manager name. Save it as QM variable
QM=$1

if [ -z "$QM" ]
then
  echo "ERROR! No queue manager name supplied"
  exit 1
```

```
fi

# End any queue manager processes which might be running.

srchstr="( |-m)$QM *.*$"
for process in amqzmuc0 amqzxma0 amqfcxba amqfqpub amqpcsea amqzlaa0 \
               amqzlsa0 runmqchi runmqlsr amqcrsta amqrrmfa amqrmppa \
               amqzfuma amqzmuf0 amqzmur0 amqzmgr0
 do
  ps -ef | tr "\t" " " | grep $process | grep -v grep | \
    egrep "$srchstr" | awk '{print $2}'| \
       xargs kill -9 > /dev/null 2>&1
done

# It is now safe to start the queue manager.
# The strmqm command does not use the -x flag.
strmqm ${QM}
```

Puede modificar el script para iniciar otros programas relacionados.

Linux AIX *Script de shell de ejemplo para detener un gestor de colas de clúster HA en AIX and Linux*

En la mayoría de los casos puede utilizar un script de shell para detener el gestor de colas. Aquí se proporcionan ejemplos de scripts de shell adecuados. Puede adaptarlos según sus necesidades y utilizarlos para detener el gestor de colas bajo el control del clúster HA.

El script siguiente es un ejemplo de cómo detener inmediatamente un gestor de colas sin presuponer nada sobre el estado actual del gestor de colas. El script debe ser ejecutado por el usuario mqm. Por lo tanto, podría ser necesario acomodar este script en un script de shell para conmutar el usuario del usuario del clúster HA a mqm. (Se proporciona un script de shell de ejemplo en "Scripts de shell de ejemplo para iniciar un gestor de colas de clúster HA en AIX and Linux" en la página 508.)

```
#!/bin/ksh
#
# The script ends the QM by using two phases, initially trying an immediate
# end with a time-out and escalating to a forced stop of remaining
# processes.
#
# The script must be run by the mqm user.
#
# There are two arguments: the queue manager name and a timeout value.
QM=$1
TIMEOUT=$2

if [ -z "$QM" ]
then
  echo "ERROR! No queue manager name supplied"
  exit 1
fi

if [ -z "$TIMEOUT" ]
then
  echo "ERROR! No timeout specified"
  exit 1
fi

for severity in immediate brutal
do
  # End the queue manager in the background to avoid
  # it blocking indefinitely. Run the TIMEOUT timer
  # at the same time to interrupt the attempt, and try a
  # more forceful version. If the brutal version fails,
  # nothing more can be done here.

  echo "Attempting ${severity} end of queue manager '${QM}'"
  case $severity in

  immediate)
    # Minimum severity of endmqm is immediate which severs connections.
    # HA cluster should not be delayed by clients
    endmqm -i ${QM} &
    ;;

  brutal)
```

```
    # This is a forced means of stopping queue manager processes.

    srchstr="( |-m)$QM *.*$"
    for process in amqzmuc0 amqzxma0 amqfcxba amqfqpub amqpcsea amqzlaa0 \
                amqzlsa0 runmqchi runmqlsr amqcrsta amqrrmfa amqrmppa \
                amqzfuma amqzmuf0 amqzmur0 amqzmgr0
    do
      ps -ef | tr "\t" " " | grep $process | grep -v grep | \
          egrep "$srchstr" | awk '{print $2}'| \
             xargs kill -9 > /dev/null 2>&1
    done

  esac

  TIMED_OUT=yes
  SECONDS=0
  while (( $SECONDS < ${TIMEOUT} ))
  do
   TIMED_OUT=yes
   i=0
   while [ $i -lt 5 ]
   do
     # Check for execution controller termination
     srchstr="( |-m)$QM *.*$"
     cnt=`ps -ef | tr "\t" " " | grep amqzxma0 | grep -v grep | \
       egrep "$srchstr" | awk '{print $2}' | wc -l `
     i=`expr $i + 1`
     sleep 1
     if [ $cnt -eq 0 ]
     then
       TIMED_OUT=no
       break
     fi
   done

   if [ ${TIMED_OUT} = "no" ]
   then
     break
   fi

   echo "Waiting for ${severity} end of queue manager '${QM}'"
   sleep 1
  done # timeout loop

  if [ ${TIMED_OUT} = "yes" ]
  then
    continue        # to next level of urgency
  else
    break           # queue manager is ended, job is done
  fi

done # next phase
```

Nota: En función de los procesos que se están ejecutando para un gestor de colas específico, la lista de procesos del gestor de colas incluida en este script podría no estar completa o podría incluir más procesos de los que se están ejecutando para dicho gestor de colas:

```
for process in amqzmuc0 amqzxma0 amqfcxba amqfqpub amqpcsea amqzlaa0 \
               amqzlsa0 runmqchi runmqlsr amqcrsta amqrrmfa amqrmppa \
               amqzfuma amqzmuf0 amqzmur0 amqzmgr0
```

Un proceso se puede incluir o excluir de esta lista basándose en qué características está configurada y qué procesos se están ejecutando para un gestor de colas específico. Para obtener una lista completa de procesos e información sobre cómo detener los procesos en un orden específico, consulte Detener un gestor de colas manualmente en UNIX y Linux.

Linux AIX *Supervisión de un gestor de colas de clúster HA en AIX and Linux*

Es habitual ofrecer un método para que el clúster de alta disponibilidad (HA) supervise periódicamente el estado del gestor de colas. En la mayoría de los casos, una de las soluciones es utilizar un script de shell. Aquí se proporcionan ejemplos de scripts de shell adecuados. Puede personalizar estos scripts según sus necesidades y utilizarlos para realizar comprobaciones de supervisión adicionales específicas de su entorno.

Es posible tener varias instalaciones de IBM MQ que coexistan en un sistema. Para obtener más información sobre varias instalaciones, consulte Varias instalaciones. Si tiene previsto utilizar el script de supervisión en varias instalaciones, es posible que tenga que realizar algunos pasos adicionales. Si tiene una instalación primaria, no es necesario que especifique *MQ_INSTALLATION_PATH* para utilizar el script. De lo contrario, utilice los pasos siguientes para asegurarse de que el *MQ_INSTALLATION_PATH* se ha identificado correctamente:

1. Utilice el mandato **crtmqenv** de una instalación de IBM MQ para identificar la *MQ_INSTALLATION_PATH* correcta para un gestor de colas:

   ```
   crtmqenv -m qmname
   ```

   Este mandato devuelve el valor de *MQ_INSTALLATION_PATH* correcta para el gestor de colas especificado por *nombreGC*.
2. Ejecute el script de supervisión con los parámetros *nombreGC* y *MQ_INSTALLATION_PATH* adecuados.

Nota: PowerHA para AIX no proporciona una forma de suministrar un parámetro al programa de supervisión para el gestor de colas. Debe crear un programa de supervisión distinto para cada gestor de colas, que encapsule el nombre del gestor de colas. El siguiente es un ejemplo de un script utilizado en AIX para encapsular el nombre del gestor de colas:

```
#!/bin/ksh
su mqm -c name_of_monitoring_script qmname  MQ_INSTALLATION_PATH
```

donde *MQ_INSTALLATION_PATH* es un parámetro opcional que especifica la vía de acceso a la instalación de IBM MQ a la que está asociado el gestor de colas *nombreGC*.

El siguiente script tiene posibilidades de que **runmqsc** se cuelgue. Normalmente, los clústeres HA manejan un script de supervisión que se cuelga como un error y son de por sí poco propicios a esta posibilidad.

Sin embargo, el script acepta que el gestor de colas está en el estado de inicio. Esto se debe a que el clúster HA empieza a supervisar el gestor de colas en cuanto lo ha iniciado. Algunos clústeres HA distinguen entre una fase de inicio y una fase de ejecución de los recursos, pero es necesario configurar la duración de la fase de inicio. Puesto que el tiempo que se necesita para iniciar un gestor de colas depende de la cantidad de trabajo que deber realizar, es difícil elegir el tiempo máximo que un gestor de colas requiere para iniciarse. Si elige un valor demasiado bajo, el clúster HA presupone incorrectamente que el gestor de colas ha fallado porque no ha completado su inicio. Esto podría generar una secuencia infinita de anomalías.

Este script lo debe ejecutar el usuario mqm; por consiguiente, es necesario acomodar el script en un script de shell para que el usuario cambie del usuario de clúster HA a mqm (en “Scripts de shell de ejemplo para iniciar un gestor de colas de clúster HA en AIX and Linux” en la página 508) se proporciona un script de shell de ejemplo:

```
#!/bin/ksh
#
# This script tests the operation of the queue manager.
#
# An exit code is generated by the runmqsc command:
# 0  => Either the queue manager is starting or the queue manager is running and responds.
#       Either is OK.
# >0 => The queue manager is not responding and not starting.
#
# This script must be run by the mqm user.
QM=$1
MQ_INSTALLATION_PATH=$2

if [ -z "$QM" ]
then
  echo "ERROR! No queue manager name supplied"
  exit 1
fi

if [ -z "$MQ_INSTALLATION_PATH" ]
```

```
then
  # No path specified, assume system primary install or MQ level < 7.1.0.0
  echo "INFO: Using shell default value for MQ_INSTALLATION_PATH"
else
  echo "INFO: Prefixing shell PATH variable with $MQ_INSTALLATION_PATH/bin"
  PATH=$MQ_INSTALLATION_PATH/bin:$PATH
fi

# Test the operation of the queue manager. Result is 0 on success, non-zero on error.
echo "ping qmgr" | runmqsc ${QM} > /dev/null 2>&1
pingresult=$?

if [ $pingresult -eq 0 ]
then # ping succeeded

  echo "Queue manager '${QM}' is responsive"
  result=0

else # ping failed

  # Don't condemn the queue manager immediately, it might be starting.
  srchstr="( |-m)$QM *.*$"
  cnt=`ps -ef | tr "\t" " " | grep strmqm | grep "$srchstr" | grep -v grep \
                | awk '{print $2}' | wc -l`
  if [ $cnt -gt 0 ]
  then
    # It appears that the queue manager is still starting up, tolerate
    echo "Queue manager '${QM}' is starting"
    result=0
  else
    # There is no sign of the queue manager starting
    echo "Queue manager '${QM}' is not responsive"
    result=$pingresult
  fi

fi

exit $result
```

Linux AIX

Colocación del gestor de colas bajo el control de clúster HA en AIX and Linux

Debe configurar el gestor de colas, bajo el control del clúster HA, con la dirección IP y los discos compartidos del gestor de colas.

Acerca de esta tarea

Para poner el gestor de colas bajo control del clúster HA, debe definir un grupo de recursos para que contenga el gestor de colas y todos sus recursos asociados.

Procedimiento

1. Cree el grupo de recursos que contiene el gestor de colas, el volumen o el grupo de discos del gestor de colas y la dirección IP del gestor de colas.

   La dirección IP es una dirección IP virtual y no la dirección IP del sistema.
2. Verifique que el clúster HA conmuta correctamente los recursos entre los nodos del clúster y está preparado para controlar el gestor de colas.

Linux AIX

Supresión de un gestor de colas de clúster HA en AIX and Linux

Si lo desea puede eliminar un gestor de colas de un nodo que ya no es necesario para ejecutar el gestor de colas.

Acerca de esta tarea

Para eliminar el gestor de colas de un nodo de un clúster HA, debe eliminar la información de configuración.

Procedimiento

1. Elimine el nodo del clúster HA de modo que el clúster HA ya no intente activar el gestor de colas en este nodo.
2. Utilice el siguiente mandato **rmvmqinf** para eliminar la información de configuración del gestor de colas:

   ```
   rmvmqinf qmgrname
   ```
3. Opcional: Para suprimir completamente el gestor de colas, utilice el mandato **dltmqm**.

   Importante: Tenga en cuenta que al suprimir el gestor de colas mediante el mandato **dltmqm**, se suprimen completamente los datos y los archivos de registro del gestor de colas.

   Cuando haya suprimido el gestor de colas, podrá utilizar el mandato **rmvmqinf** para eliminar la información de configuración restante de los otros nodos.

Windows Soporte de Microsoft Cluster Service (MSCS)

Introducción y configuración de MSCS para dar soporte a la sustitución por anomalía de servidores virtuales. MSCS también se conoce como Windows Server Failover Clustering (WSFC).

Esta información sólo se aplica a IBM MQ for Windows.

Nota: A partir de Windows Server 2016, el nuevo nombre para Microsoft Cluster Service (MSCS) es Windows Server Failover Clustering (WSFC).

MSCS/WSFC le permite conectar servidores a un clúster, lo que proporciona una mayor disponibilidad de datos y aplicaciones, y facilita la gestión del sistema. MSCS/WSFC puede detectar y recuperarse automáticamente de errores de servidor o aplicación.

MSCS/WSFC da soporte a la migración tras error de servidores virtuales, que corresponden a aplicaciones, sitios web, colas de impresión o comparticiones de archivos (incluidos, por ejemplo, sus husillos de disco, archivos y direcciones IP).

Migración tras error es el proceso mediante el cual MSCS/WSFC detecta una anomalía en una aplicación en un sistema del clúster y concluye la aplicación interrumpida de forma ordenada, transfiere sus datos de estado al otro sistema y vuelve a iniciar la aplicación allí.

Para obtener información sobre cómo configurar y utilizar clústeres de migración tras error, consulte los subtemas.

Windows *Introducción a los clústeres de MSCS*

Los clústeres de Microsoft Cluster Service (MSCS) son grupos de dos o más sistemas, conectados entre sí y configurados de tal forma que, si uno falla, MSCS realiza una *migración tras error*, transfiriendo los datos de estado de las aplicaciones del sistema anómalo a otro sistema del clúster y reiniciando su operación allí.

Nota: A partir de Windows Server 2016, el nuevo nombre para Microsoft Cluster Service (MSCS) es Windows Server Failover Clustering (WSFC).

"Configuraciones de alta disponibilidad" en la página 501 contiene una comparación entre los clústeres de MSCS, los gestores de colas multiinstancia y los clústeres de IBM MQ.

En esta sección y sus temas subordinados, el término *clúster*, cuando se utiliza por sí mismo, **siempre** significa un clúster de MSCS. Es diferente de un clúster de IBM MQ, que se describe en otra parte de esta guía.

Un clúster de dos máquinas consta de dos sistemas (por ejemplo, A y B) que se interconectan a una red para acceso de cliente mediante una *dirección IP virtual*. También se pueden conectar entre sí mediante una o varias redes privadas. En cada uso, A y B comparten como mínimo un disco para las aplicaciones de servidor. También hay otro disco compartido, que debe ser una matriz redundante de discos independientes (*RAID*) Nivel 1, para uso exclusivo de MSCS; esto se conoce como disco de *quórum*. Los monitores MSCS de ambos sistemas comprueba que el hardware y el software se ejecuten correctamente.

En una configuración sencilla como esta, ambos sistemas tienen todas las aplicaciones instaladas en el mismo, pero solamente el sistema A se ejecuta con aplicaciones activas, mientras que el sistema B simplemente está en ejecución y a la espera. Si el sistema A sufre cualquier problema de una serie de problemas, MSCS concluye la aplicación que se ha interrumpido de forma ordenada, transfiere sus datos de estado a otro sistema y reinicia allí la aplicación. Esto se conoce como *sustitución por anomalía*. Se puede hacer que las aplicaciones estén *preparadas para el clúster*, de modo que puedan interactuar de forma completa con MSCS y puedan ejecutar la sustitución por anomalía correctamente.

En la Figura 71 en la página 514 aparece una configuración típica de un clúster de dos sistemas.

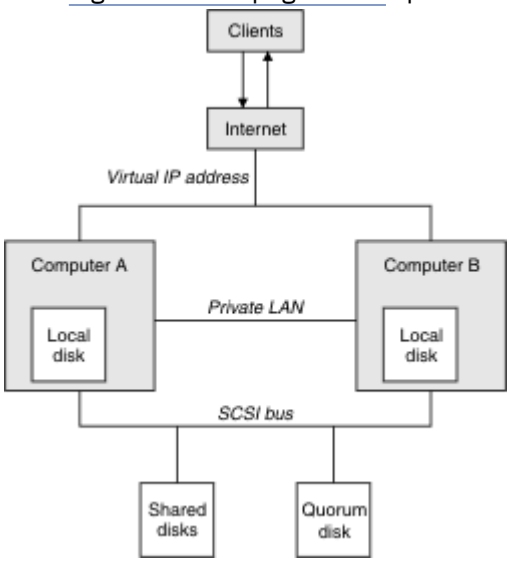


Figura 71. Clúster de MSCS de dos sistemas

Bajo el control de MSCS, cada sistema puede acceder al disco compartido pero sólo uno cada vez. Si se produce una sustitución por anomalía, MSCS pasa el acceso al otro sistema. El disco compartido propiamente dicho suele ser un RAID, pero no necesariamente debe serlo.

Cada sistema está conectado a la red externa para acceso de cliente y cada uno tiene una dirección IP. Sin embargo, un cliente externo que se comunique con este clúster solamente advierte una *dirección IP virtual* y MSCS direccionará el tráfico IP dentro del clúster que corresponda.

MSCS también efectúa sus propias comunicaciones entre dos sistemas, ya sea a través de una o varias conexiones privadas o a través de la red pública, por ejemplo para supervisar sus estados mediante la pulsación, para sincronizar sus bases de datos.

Windows *Configuración de IBM MQ para la agrupación en clúster de MSCS*

Puede configurar IBM MQ para la agrupación en clúster haciendo que el gestor de colas sea la unidad de sustitución por anomalía para MSCS. Debe definir un gestor de colas como un recurso para MSCS y éste podrá supervisarlo y transferirlo a otro sistema del clúster si se produce un problema.

Nota: A partir de Windows Server 2016, el nuevo nombre para Microsoft Cluster Service (MSCS) es Windows Server Failover Clustering (WSFC).

Para configurar el sistema para este fin, empiece por instalar IBM MQ en cada sistema del clúster.

Como el gestor de colas está asociado con el nombre de instalación de IBM MQ, el nombre de instalación de IBM MQ en todos los sistemas del clúster deben ser el mismo. Consulte Instalación y desinstalación.

Los gestores de colas propiamente dichos sólo necesitan existir en el sistema en que los crea. Si se produce una sustitución por anomalía, MSCS inicia los gestores de colas en el otro sistema. Sin embargo, los gestores de colas deben tener sus archivos de anotaciones y de datos en un disco compartido del clúster y no en una unidad local. Si ya tiene instalado un gestor de colas en una unidad local, puede migrarlo utilizando una herramienta que se proporciona con IBM MQ; consulte “Mover un gestor de colas al almacenamiento de MSCS” en la página 517. Si desea crear nuevos gestores de colas para utilizarlos con MSCS, consulte “Creación de un gestor de colas para utilizarlo con MSCS” en la página 516.

Tras la instalación y la migración, utilice el Administrador de clústeres de MSCS para hacer que MSCS reconozca sus gestores de colas; consulte “Poner un gestor de colas bajo control de MSCS” en la página 518.

Si decide eliminar un gestor de colas del control de MSCS, utilice el procedimiento que se describe en “Eliminar un gestor de colas del control de MSCS” en la página 525.

Windows *Configuración de la simetría y MSCS*

Cuando una aplicación pasa de un nodo a otro debe comportarse del mismo modo en cualquiera de los nodos. La mejor manera de garantizar esto es hacer que los entornos sean idénticos.

Si puede, configure un clúster que tenga hardware, software de sistema operativo, software de productos y configuración idénticos. En concreto, asegúrese de que todo el software necesario instalado en los dos equipos es idéntico en términos de versión, mantenimiento, nivel, SupportPacs, vías de acceso y salidas y que hay un espacio de nombres común (entorno de seguridad) tal como se describe en “Seguridad MSCS” en la página 515.

Windows *Seguridad MSCS*

Para que la seguridad MSCS funcione correctamente, siga estas indicaciones.

Las directrices son las siguientes:

- Asegúrese de que tiene instalaciones de software idénticas en cada sistema del clúster.
- Cree un espacio de nombres común (entorno de seguridad) en todo el clúster.
- Defina los nodos del clúster de MSCS como miembros de un dominio, dentro del cual la cuenta de usuario que sea el *propietario del clúster* sea una cuenta de dominio.
- Defina las otras cuentas de usuario del clúster como cuentas de dominio también, para que estén disponibles en ambos nodos. Esto es así automáticamente si ya tiene un dominio, y las cuentas asociadas a IBM MQ son cuentas de dominio. Si actualmente no tiene un dominio, puede optar por definir un *minidominio* para atender los nodos del clúster y las cuentas asociadas. El objetivo es hacer que el clúster de dos sistemas parezca un solo recurso del sistema.

  Recuerde que una cuenta que es local para un sistema no existe en el otro. Incluso si crea una cuenta con el mismo nombre en el otro sistema, su identificador de seguridad (SID) será diferente, por lo tanto, cuando su aplicación se traslade al otro nodo, los permisos no existirán en dicho nodo.

Durante una sustitución por anomalía o un traslado, el soporte MSCS de IBM MQ garantiza que todos los archivos que contienen objetos de gestor de colas tengan permisos equivalentes en el nodo de destino. Explícitamente, el código comprueba que los grupos Administradores y mqm, y la cuenta SYSTEM, tienen control completo y que si `Everyone` tenía acceso de lectura en el nodo antiguo, ese permiso se añade en el nodo de destino.

Puede utilizar una cuenta de dominio para ejecutar el servicio IBM MQ. Asegúrese de que exista en el grupo mqm local de cada sistema del clúster.

Windows *Utilización de varios gestores de colas con MSCS*

Si está ejecutando más de un gestor de colas en un sistema, puede elegir una de estas configuraciones.

Las configuraciones son las siguientes:

- Todos los gestores de colas en un único grupo. En esta configuración, si surge un problema con algún gestor de colas, se produce una sustitución por anomalía y todos los gestores de colas del grupo pasan al otro sistema como un grupo.
- Un solo gestor de colas en cada grupo. En esta configuración, si surge un problema con el gestor de colas, éste es el único que se pasa al otro sistema cuando se produce la sustitución por anomalía, sin afectar a los demás gestores de colas.
- Una combinación de las dos primeras configuraciones.

Windows *Modalidades de clúster y MSCS*

Hay dos modalidades en las que puede ejecutar un sistema de clúster con IBM MQ en Windows: Activa/Pasiva o Activa/Activa.

Nota: Si utiliza MSCS junto con Microsoft Transaction Server (COM+), no puede utilizar la modalidad Activa/Activa.

Modalidad Activa/Pasiva

En modalidad Activa/Pasiva, el sistema A tiene la aplicación en ejecución y el sistema B es el sistema de reserva, que sólo se utiliza cuando MSCS detecta un problema.

Puede utilizar esta modalidad con un solo disco compartido pero, si alguna aplicación provoca una anomalía, **todas** las aplicaciones deben transferirse como un grupo (porque sólo un sistema puede acceder al disco compartido simultáneamente).

Puede configurar MSCS con A como el sistema *preferido*. Así, cuando el sistema A haya sido reparado o reemplazado y vuelva a funcionar correctamente, MSCS lo detectará y cambiará automáticamente la aplicación al sistema A.

Si ejecuta más de un gestor de colas, considere la posibilidad de tener un disco compartido independiente para cada uno. Coloque después cada gestor de colas en un grupo separado en MSCS. De esta manera, cualquier gestor de colas puede realizar la sustitución por anomalía al otro sistema sin afectar al resto de gestores de colas.

Modalidad Activa/Activa

En modalidad Activa/Activa, los sistemas A y B tienen ambos aplicaciones en ejecución y los grupos que hay en cada sistema tienen definido el otro sistema como sistema de reserva. Si se detecta un error en el sistema A, MSCS transfiere los datos de estado al sistema B y reinicia la aplicación allí. El sistema B ejecuta entonces su propia aplicación y la del sistema A.

Para esta configuración debe tener al menos dos disco compartidos. Puede configurar MSCS con A como el sistema preferido para las aplicaciones de A, y B como el sistema preferido para las aplicaciones de B. Después de la sustitución por anomalía y la reparación, cada aplicación termina volviendo automáticamente a su propio sistema.

Para IBM MQ, esto significa que podría, por ejemplo, ejecutar dos gestores de colas, uno en cada sistema A y B, cada uno de ellos aprovechando toda la potencia de su propio sistema. Después de una anomalía en el sistema A, ambos gestores de colas se ejecutarán en el sistema B. Esto supondrá compartir la potencia de un ordenador, con una capacidad reducida para procesar grandes cantidades de datos a gran velocidad. No obstante, las aplicaciones más importantes seguirán estando disponibles mientras encuentra y repara la anomalía del sistema A.

Windows *Creación de un gestor de colas para utilizarlo con MSCS*

Este procedimiento garantiza que se crea un nuevo gestor de colas de forma que sea adecuado para preparar y colocar bajo el control de Microsoft Cluster Service (MSCS).

Nota: A partir de Windows Server 2016, el nuevo nombre para Microsoft Cluster Service (MSCS) es Windows Server Failover Clustering (WSFC).

Puede comenzar por crear el gestor de colas con todos sus recursos en una unidad local y luego migre los archivos de anotaciones y los archivos de datos a un disco compartido. Puede invertir esta operación. **No** intente crear un gestor de colas con sus recursos en una unidad compartida.

Puede crear un gestor de colas para utilizarlo con MSCS de dos maneras, desde un indicador de mandatos o en IBM MQ Explorer. La ventaja de utilizar un indicador de mandatos es que el gestor de colas se crea *detenido* y se establece en *inicio manual*, con lo que está preparado para MSCS. (IBM MQ Explorer inicia automáticamente un nuevo gestor de colas y lo establece en inicio automático después de la creación. Debe modificar esto.)

Creación de un gestor de colas desde un indicador de mandatos

Siga estos pasos para crear un gestor de colas desde un indicador de mandatos, para su uso con MSCS:

1. Asegúrese de que tiene la variable de entorno `MQSPREFIX` establecida para hacer referencia a una unidad local, por ejemplo `C:\IBM MQ`. Si la modifica, reinicie la máquina para que el sistema pueda aplicar el cambio. Si no establece la variable, el gestor de colas se crea en el directorio predeterminado de IBM MQ para gestores de colas.
2. Cree el gestor de colas mediante el mandato **crtmqm**. Por ejemplo, para crear un gestor de colas denominado `mscs_test` en el directorio predeterminado, utilice:

   ```
   crtmqm mscs_test
   ```

3. Continúe en el apartado "Mover un gestor de colas al almacenamiento de MSCS" en la página 517.

Creación de un gestor de colas utilizando IBM MQ Explorer

Siga estos pasos para crear un gestor de colas utilizando IBM MQ Explorer, para su uso con MSCS:

1. Inicie IBM MQ Explorer desde el menú Inicio.
2. En la vista Navegador, expanda los nodos de árbol para localizar el nodo de árbol `Gestores de colas`.
3. Pulse con el botón derecho en el nodo de árbol `Gestores de colas` y seleccione **Nuevo** > **Gestor de colas**. Aparece el panel Crear gestor de colas.
4. Complete el diálogo (Paso 1) y, a continuación, pulse **Siguiente>**.
5. Complete el diálogo (Paso 2) y, a continuación, pulse **Siguiente>**.
6. Complete el diálogo (Paso 3), asegurándose de que las opciones `Iniciar gestor de colas` y `Crear canal de conexión de servidor` no están seleccionadas y, a continuación, pulse **Siguiente>**.
7. Complete el diálogo (Paso 4) y pulse **Finalizar**.
8. Continúe en el apartado "Mover un gestor de colas al almacenamiento de MSCS" en la página 517.

Windows *Mover un gestor de colas al almacenamiento de MSCS*

Este procedimiento configura un gestor de colas existente para que sea adecuado para colocarlo bajo el control de Microsoft Cluster Service (MSCS).

Para realizarlo, coloque los archivos de anotaciones y los archivos de datos en discos compartidos, para que de este modo estén disponibles para el otro sistema en caso de que se produzca una anomalía. Por ejemplo, el gestor de colas existente puede tener vías de acceso como `C:\WebSphere MQ\log\`*`QMname`* y `C:\WebSphere MQ\qmgrs\`*`QMname`*.

Atención: No intente mover los archivos a mano; utilice el programa de utilidad que se proporciona como parte del soporte de MSCS de IBM MQ como se describe en este tema.

Si el gestor de colas que está moviendo utiliza conexiones TLS y el repositorio de claves TLS se encuentra en el directorio de datos del gestor de colas de la máquina local, el repositorio de claves se moverá con el gestor de colas al disco compartido. De forma predeterminada, el atributo del gestor de colas que especifica la ubicación del repositorio de claves TLS, SSLKEYR, se establece en *MQ_INSTALLATION_PATH*`\qmgrs\QMGRNAME\ssl\key`, que está bajo el directorio de datos del gestor de colas. *MQ_INSTALLATION_PATH* representa el directorio de alto nivel en el que está instalado IBM MQ. El mandato `hamvmqm` no modifica este atributo de gestor de colas. En esta situación debe modificar el atributo de gestor de colas, SSLKEYR, utilizando IBM MQ Explorer o el mandato de MQSC `ALTER QMGR`, para apuntar el nuevo archivo de repositorio de claves de TLS.

Nota: A partir de Windows Server 2016, el nuevo nombre para Microsoft Cluster Service (MSCS) es Windows Server Failover Clustering (WSFC).

El procedimiento es el siguiente:

1. Concluya el gestor de colas y compruebe que no haya errores.
2. Si los archivos de anotaciones o los archivos de colas del gestor de colas ya están almacenados en un disco compartido, puede ignorar el resto de este procedimiento y pasar directamente al apartado “Poner un gestor de colas bajo control de MSCS” en la página 518.
3. Efectúe una copia de seguridad de soportes completa de los archivos de colas y de los archivos de registro y guarde la copia de seguridad en un lugar seguro (consulte “Archivos de anotaciones del gestor de colas” en la página 528 si desea saber por qué esto es importante).
4. Si ya tiene un recurso de disco compartido adecuado, continúe en el paso 6. De lo contrario, utilice el Administrador de clústeres de MSCS para crear un recurso de tipo *disco compartido* con capacidad suficiente para almacenar los archivos de registro del gestor de colas y los archivos de datos (cola).
5. Pruebe el disco compartido utilizando el Administrador de clústeres de MSCS para trasladarlo de un nodo de clúster al otro y otra vez al primero.
6. Asegúrese de que el disco compartido está en línea en el nodo del clúster donde se almacenan localmente los archivos de anotaciones y de datos.
7. Ejecute el programa de utilidad para mover el gestor de colas como se indica a continuación:

   ```
   hamvmqm /m qmname /dd " e: \
   IBM MQ " /ld " e: \
   IBM MQ \log"
   ```

   sustituyendo *qmname* por el nombre del gestor de colas, *e* por la letra de unidad de disco compartido y *IBM MQ* por el directorio elegido. Los directorios se crean si todavía no existen.
8. Pruebe el gestor de colas para asegurarse de que funciona, utilizando IBM MQ Explorer. Por ejemplo:
   a. Pulse el botón derecho del ratón en el nodo de árbol del gestor de colas y seleccione **Iniciar**. El gestor de colas se inicia.
   b. Pulse con el botón derecho del ratón en el nodo de árbol `Colas` y, a continuación, seleccione **Nueva** > **Cola local ...**, y asigne un nombre a la cola.
   c. Pulse **Finalizar**.
   d. Pulse el botón derecho del ratón en la cola y seleccione **Transferir mensaje de prueba...**. Aparece el panel Transferir mensaje de prueba.
   e. Escriba el texto del mensaje, luego pulse **Transferir mensaje de prueba** y cierre el panel.
   f. Pulse el botón derecho del ratón en la cola y seleccione **Examinar mensajes...**. Aparece el panel Examinador de mensajes.
   g. Asegúrese de que su mensaje está en la cola y pulse **Cerrar**. El panel Examinador de mensajes se cierra.
   h. Pulse el botón derecho del ratón en la cola y seleccione **Borrar mensajes...**. Se borran los mensajes de la cola.
   i. Pulse el botón derecho del ratón en la cola y seleccione **Suprimir...**. Aparece un panel de confirmación, pulse **Aceptar**. Se suprime la cola.
   j. Pulse el botón derecho del ratón en el nodo de árbol del gestor de colas y luego seleccione **Detener...**. Aparece el panel Finalizar el gestor de colas.
   k. Pulse **Aceptar**. El gestor de colas se detiene.
9. Como administrador de IBM MQ, asegúrese de que el atributo de inicio del gestor de colas esté establecido en manual. En IBM MQ Explorer, establezca el campo de Inicio en `manual` en el panel de propiedades de gestor de colas.
10. Continúe en el apartado “Poner un gestor de colas bajo control de MSCS” en la página 518.

Windows *Poner un gestor de colas bajo control de MSCS*

Cómo colocar un gestor de colas bajo el control de Microsoft Cluster Service (MSCS), incluidas las tareas de requisito previo.

Nota: A partir de Windows Server 2016, el nuevo nombre para Microsoft Cluster Service (MSCS) es Windows Server Failover Clustering (WSFC).

Antes de poner un gestor de colas bajo el control MSCS/WSFC

Antes de colocar un gestor de colas bajo el control MSCS/WSFC, realice los pasos siguientes:

1. Asegúrese de que IBM MQ y su soporte MSCS/WSFC estén instalados en ambas máquinas del clúster y que el software de cada sistema sea idéntico, tal como se describe en "Configuración de IBM MQ para la agrupación en clúster de MSCS" en la página 514.
2. Utilice el programa de utilidad **haregtyp** para registrar IBM MQ como un tipo de recurso MSCS en todos los nodos del clúster. Consulte "Soporte para programas de utilidad MSCS" en la página 529
3. Si todavía no lo ha hecho, cree un gestor de colas para utilizarlo con MSCS/WSFC.
4. Si ya ha creado el gestor de colas o si ya existe, asegúrese de que ha llevado a cabo el procedimiento de "Mover un gestor de colas al almacenamiento de MSCS" en la página 517.
5. Si el gestor de colas se está ejecutando, deténgalo utilizando un indicador de mandatos o IBM MQ Explorer.
6. Pruebe el funcionamiento de MSCS/WSFC de las unidades compartidas antes de continuar con cualquiera de los siguientes procedimientos de Windows en este tema.

Windows Server 2012, 2016, 2019 o 2022

Para colocar un gestor de colas bajo el control MSCS/WSFC en Windows Server 2012 o posterior, utilice el procedimiento siguiente:

1. Inicie la sesión en el sistema del nodo del clúster que alberga el gestor de colas o inicie la sesión en una estación de trabajo remota como usuario con permisos de administración del clúster y conéctese en el nodo del clúster que alberga el gestor de colas.
2. Inicie la herramienta Administración de clúster de conmutación por error.
3. Pulse con el botón derecho del ratón en **Gestión de clúster de migración tras error > Conectar clúster ...** para abrir una conexión con el clúster.
4. A diferencia del esquema de grupo utilizado en el Administrador de clústeres de MSCS en versiones anteriores de Windows, la herramienta Administración de clúster de conmutación por error emplea el concepto de servicios y aplicaciones. Un servicio o una aplicación configurados contienen todos los recursos necesarios para que un aplicación se agrupe en clúster. Puede configurar un gestor de colas bajo WSFC de la forma siguiente:
   a. Pulse el clúster con el botón derecho y seleccione **Configurar rol** para iniciar el asistente de configuración.
   b. Seleccione **Otro servidor** en el panel **Seleccionar servicio o aplicación**.
   c. Seleccione una dirección IP apropiada como punto de acceso de cliente.

      Esta dirección debe ser una dirección IP no utilizada que los clientes y otros gestores de colas utilizarán para conectarse con el gestor de colas *virtual*. Esta dirección IP no es la dirección normal (estática) de cada nodo; es una dirección adicional que *flota* entre ellos. Aunque WSFC maneja el direccionamiento de esta dirección, **no** verifica que se pueda acceder a la dirección.
   d. Asigne un dispositivo de almacenamiento para el uso exclusivo del gestor de colas. Este dispositivo se tiene crear como una instancia de recurso para poderse asignar.

      Puede utilizar una unidad para almacenar los archivos de registro y de colas o puede dividirlos en varias unidades. En cualquiera de los casos, si cada gestor de colas tiene su propio disco compartido, asegúrese de que todas las unidades que utiliza este gestor de colas sean exclusivas a este gestor de colas, es decir, que ningún otro recurso necesite las unidades. Asegúrese también de que crea una instancia de recurso para cada unidad que utiliza el gestor de colas.

      El tipo de recurso de una unidad depende del soporte SCSI que esté utilizando; consulte las instrucciones del adaptador SCSI. Es posible que ya haya grupos y recursos para cada una de

las unidades compartidas. Si es así, no hace falta crear la instancia de recurso para cada unidad. Muévala del grupo actual al que ha creado para el gestor de colas.

Para cada recurso de unidad, establezca los propietarios posibles para ambos nodos. Establezca los recursos dependientes en none (ninguno).

e. Seleccione el recurso **MQSeries MSCS** en el panel **Seleccionar tipo de recurso**.

f. Complete los pasos restantes del asistente.

5. Antes de poner el recurso en línea, el recurso MQSeries MSCS precisa de configuración adicional:

a. Seleccione el nuevo servicio definido que contiene un recurso llamado 'Nuevo MQSeries MSCS'.

b. Pulse con el botón derecho del ratón en **Propiedades** en el recurso IBM MQ .

c. Configure el recurso:

- `Name` elija un nombre que facilite la identificación del gestor de colas para el que se encuentra.
- `Run in a separate Resource Monitor` para un mejor aislamiento
- `Possible owners` establecer ambos nodos
- `Dependencies` añadir la unidad y la dirección IP para este gestor de colas.

  Aviso: No añadir estas dependencias significa que IBM MQ intenta grabar el estado del gestor de colas en el disco en clúster incorrecto durante las sustituciones por anomalía. Dado que puede haber muchos procesos que estén intentando grabar en este disco simultáneamente, algunos procesos de IBM MQ podrían bloquearse para impedir su ejecución.
- `Parameters` como sigue:
  - `QueueManagerName` (obligatorio); el nombre del gestor de colas que este recurso va a controlar. Este gestor de colas debe existir en el sistema local.
  - `PostOnlineCommand` (opcional); puede especificar un programa para ejecutarlo cuando el estado del recurso de gestor de colas pase de fuera de línea a en línea. Si desea ver información más detallada, consulte "PostOnlineCommand y PreOfflineCommand en MSCS" en la página 528.
  - `PreOfflineCommand` (opcional); puede especificar un programa para ejecutarlo cuando el estado del recurso de gestor de colas pase de en línea a fuera de línea. Si desea ver información más detallada, consulte "PostOnlineCommand y PreOfflineCommand en MSCS" en la página 528.

    Nota: El intervalo de sondeo *looksAlive* se establece en el valor predeterminado de 5000 ms. El intervalo de sondeo *isAlive* se establece en el valor predeterminado de 60000 ms. Estos valores predeterminados sólo se pueden modificar una vez que se ha completado la definición del recurso. Para obtener más información, consulte "Sondeo de looksAlive y isAlive en MSCS" en la página 525.

d. Opcionalmente, establezca un nodo preferido (pero tome nota de los comentarios en "Utilización de nodos preferidos en MSCS" en la página 529)

e. La *Política de sustitución por anomalía* se establece de forma predeterminada en valores sensatos, pero se pueden ajustar los umbrales y los períodos que controlan *Sustitución por anomalía de recurso* y *Sustitución por anomalía de grupo* para que coincidan con las cargas impuestas en el gestor de colas.

6. Compruebe el gestor de colas pasándolo a en línea en el Administrador de clústeres de MSCS y sometiéndolo a una prueba de carga de trabajo. Si está experimentando con un gestor de colas de prueba, utilice IBM MQ Explorer. Por ejemplo:

a. Pulse con el botón derecho del ratón en el nodo de árbol `Colas` y, a continuación, seleccione **Nueva** > **Cola local ...**, y asigne un nombre a la cola.

b. Pulse **Finalizar**. Se crea la cola y aparece en la vista de contenido.

c. Pulse el botón derecho del ratón en la cola y seleccione **Transferir mensaje de prueba...**. Aparece el panel Transferir mensaje de prueba.

d. Escriba el texto del mensaje, luego pulse **Transferir mensaje de prueba** y cierre el panel.

e. Pulse el botón derecho del ratón en la cola y seleccione **Examinar mensajes...**. Aparece el panel Examinador de mensajes.

f. Asegúrese de que el mensaje está en la cola y a continuación pulse **Cerrar**. El panel Examinador de mensajes se cierra.

g. Pulse el botón derecho del ratón en la cola y seleccione **Borrar mensajes...**. Se borran los mensajes de la cola.

h. Pulse el botón derecho del ratón en la cola y seleccione **Suprimir...**. Aparece un panel de confirmación, pulse **Aceptar**. Se suprime la cola.

7. Compruebe que el gestor de colas se pueda pasar a fuera de línea y volver a pasar a en línea mediante el Administrador de clústeres de MSCS.
8. Simule una sustitución por anomalía.

   En el Administrador de clústeres de MSCS, pulse con el botón derecho del ratón en el grupo que contiene el gestor de colas y seleccione `Move Group`. Esto puede tardar algunos minutos. (Si en otro momento desea mover rápidamente un gestor de colas a otro nodo, siga el procedimiento de "Mover un gestor de colas al almacenamiento de MSCS" en la página 517.) También puede pulsar con el botón derecho del ratón y seleccionar `Initiate Failure` ; la acción (reinicio local o migración tras error) depende del estado actual y de los valores de configuración.

Windows Server 2008

Para poner un gestor de colas bajo control de MSCS en Windows Server 2008, siga este procedimiento:

1. Inicie la sesión en el sistema del nodo del clúster que alberga el gestor de colas o inicie la sesión en una estación de trabajo remota como usuario con permisos de administración del clúster y conéctese en el nodo del clúster que alberga el gestor de colas.
2. Inicie la herramienta Administración de clúster de conmutación por error.
3. Pulse con el botón derecho **Gestión de clúster de migración tras error > Gestionar un clúster ...** para abrir una conexión con el clúster.
4. A diferencia del esquema de grupo utilizado en el Administrador de clústeres de MSCS en versiones anteriores de Windows, la herramienta Administración de clúster de conmutación por error emplea el concepto de servicios y aplicaciones. Un servicio o una aplicación configurados contienen todos los recursos necesarios para que un aplicación se agrupe en clúster. Puede configurar un gestor de colas bajo MSCS del modo siguiente:

   a. Pulse con el botón derecho **Servicios y aplicaciones > Configurar un servicio o aplicación ...** para iniciar el asistente de configuración.

   b. Seleccione **Otro servidor** en el panel **Seleccionar servicio o aplicación**.

   c. Seleccione una dirección IP apropiada como punto de acceso de cliente.

      Esta dirección debe ser una dirección IP no utilizada que los clientes y otros gestores de colas utilizarán para conectarse con el gestor de colas *virtual*. Esta dirección IP no es la dirección normal (estática) de cada nodo; es una dirección adicional que *flota* entre ellos. Aunque MSCS maneja el direccionamiento de esta dirección, **no** comprueba si se ha llegado a la dirección.

   d. Asigne un dispositivo de almacenamiento para el uso exclusivo del gestor de colas. Este dispositivo se tiene crear como una instancia de recurso para poderse asignar.

      Puede utilizar una unidad para almacenar los archivos de registro y de colas o puede dividirlos en varias unidades. En cualquiera de los casos, si cada gestor de colas tiene su propio disco compartido, asegúrese de que todas las unidades que utiliza este gestor de colas sean exclusivas a este gestor de colas, es decir, que ningún otro recurso necesite las unidades. Asegúrese también de que crea una instancia de recurso para cada unidad que utiliza el gestor de colas.

      El tipo de recurso de una unidad depende del soporte SCSI que esté utilizando; consulte las instrucciones del adaptador SCSI. Es posible que ya haya grupos y recursos para cada una de

las unidades compartidas. Si es así, no hace falta crear la instancia de recurso para cada unidad. Muévala del grupo actual al que ha creado para el gestor de colas.

Para cada recurso de unidad, establezca los propietarios posibles para ambos nodos. Establezca los recursos dependientes en none (ninguno).

e. Seleccione el recurso **MQSeries MSCS** en el panel **Seleccionar tipo de recurso**.

f. Complete los pasos restantes del asistente.

5. Antes de poner el recurso en línea, el recurso MQSeries MSCS precisa de configuración adicional:

a. Seleccione el servicio recién definido que contiene un recurso denominado 'Nuevo MQSeries MSCS'.

b. Pulse el botón derecho del ratón en **Propiedades** en el recurso MQ.

c. Configure el recurso:

- `Name` elija un nombre que facilite la identificación del gestor de colas para el que se encuentra.
- `Run in a separate Resource Monitor` para un mejor aislamiento
- `Possible owners` establecer ambos nodos
- `Dependencies` añadir la unidad y la dirección IP para este gestor de colas.

  Aviso: No añadir estas dependencias significa que IBM MQ intenta grabar el estado del gestor de colas en el disco en clúster incorrecto durante las sustituciones por anomalía. Dado que puede haber muchos procesos que estén intentando grabar en este disco simultáneamente, algunos procesos de IBM MQ podrían bloquearse para impedir su ejecución.
- `Parameters` como sigue:
  - `QueueManagerName` (obligatorio); el nombre del gestor de colas que este recurso va a controlar. Este gestor de colas debe existir en el sistema local.
  - `PostOnlineCommand` (opcional); puede especificar un programa para ejecutarlo cuando el estado del recurso de gestor de colas pase de fuera de línea a en línea. Si desea ver información más detallada, consulte "PostOnlineCommand y PreOfflineCommand en MSCS" en la página 528.
  - `PreOfflineCommand` (opcional); puede especificar un programa para ejecutarlo cuando el estado del recurso de gestor de colas pase de en línea a fuera de línea. Si desea ver información más detallada, consulte "PostOnlineCommand y PreOfflineCommand en MSCS" en la página 528.

    Nota: El intervalo de sondeo *looksAlive* se establece en el valor predeterminado de 5000 ms. El intervalo de sondeo *isAlive* se establece en el valor predeterminado de 60000 ms. Estos valores predeterminados sólo se pueden modificar una vez que se ha completado la definición del recurso. Para obtener más información, consulte "Sondeo de looksAlive y isAlive en MSCS" en la página 525.

d. Opcionalmente, establezca un nodo preferido (pero tome nota de los comentarios en "Utilización de nodos preferidos en MSCS" en la página 529)

e. La *Política de sustitución por anomalía* se establece de forma predeterminada en valores sensatos, pero se pueden ajustar los umbrales y los períodos que controlan *Sustitución por anomalía de recurso* y *Sustitución por anomalía de grupo* para que coincidan con las cargas impuestas en el gestor de colas.

6. Compruebe el gestor de colas pasándolo a en línea en el Administrador de clústeres de MSCS y sometiéndolo a una prueba de carga de trabajo. Si está experimentando con un gestor de colas de prueba, utilice IBM MQ Explorer. Por ejemplo:

a. Pulse con el botón derecho del ratón en el nodo de árbol `Colas` y, a continuación, seleccione **Nueva** > **Cola local ...**, y asigne un nombre a la cola.

b. Pulse **Finalizar**. Se crea la cola y aparece en la vista de contenido.

c. Pulse el botón derecho del ratón en la cola y seleccione **Transferir mensaje de prueba...**. Se muestra el panel **Transferir mensaje de prueba**.

d. Escriba el texto del mensaje, luego pulse **Transferir mensaje de prueba** y cierre el panel.

e. Pulse el botón derecho del ratón en la cola y seleccione **Examinar mensajes...**. Se muestra el panel **Examinador de mensajes**.

f. Asegúrese de que el mensaje está en la cola y a continuación pulse **Cerrar**. Se cierra el panel **Examinador de mensajes**.

g. Pulse el botón derecho del ratón en la cola y seleccione **Borrar mensajes...**. Se borran los mensajes de la cola.

h. Pulse el botón derecho del ratón en la cola y seleccione **Suprimir...**. Aparece un panel de confirmación, pulse **Aceptar**. Se suprime la cola.

7. Compruebe que el gestor de colas se pueda pasar a fuera de línea y volver a pasar a en línea mediante el Administrador de clústeres de MSCS.
8. Simule una sustitución por anomalía.

   En el Administrador de clústeres de MSCS, pulse con el botón derecho del ratón en el grupo que contiene el gestor de colas y seleccione `Move Group`. Esto puede tardar algunos minutos. (Si en otro momento desea mover rápidamente un gestor de colas a otro nodo, siga el procedimiento de "Mover un gestor de colas al almacenamiento de MSCS" en la página 517.) También puede pulsar con el botón derecho del ratón y seleccionar `Initiate Failure` ; la acción (reinicio local o migración tras error) depende del estado actual y de los valores de configuración.

Windows 2003

Para poner un gestor de colas bajo control de MSCS en Windows 2003, siga este procedimiento:

1. Inicie la sesión en el sistema del nodo del clúster que alberga el gestor de colas o inicie la sesión en una estación de trabajo remota como usuario con permisos de administración del clúster y conéctese en el nodo del clúster que alberga el gestor de colas.
2. Inicie el Administrador de clústeres de MSCS.
3. Abra una conexión con el clúster.
4. Cree un grupo MSCS que se utilizará para contener los recursos para el gestor de colas. Asigne un nombre al grupo, de tal modo, que quede claro con qué gestor de colas está relacionado. Cada grupo puede contener varios gestores de colas, tal como se describe en "Utilización de varios gestores de colas con MSCS" en la página 515.

   Utilice el grupo para todos los pasos restantes.
5. Cree una instancia de recurso para cada unidad lógica SCSI que utilice el gestor de colas.

   Puede utilizar una unidad para almacenar los archivos de registro y de colas o puede dividirlos en varias unidades. En cualquiera de los casos, si cada gestor de colas tiene su propio disco compartido, asegúrese de que todas las unidades que utiliza este gestor de colas sean exclusivas a este gestor de colas, es decir, que ningún otro recurso necesite las unidades. Asegúrese también de que crea una instancia de recurso para cada unidad que utiliza el gestor de colas.

   El tipo de recurso de una unidad depende del soporte SCSI que esté utilizando; consulte las instrucciones del adaptador SCSI. Es posible que ya haya grupos y recursos para cada una de las unidades compartidas. Si es así, no hace falta crear la instancia de recurso para cada unidad. Muévala del grupo actual al que ha creado para el gestor de colas.

   Para cada recurso de unidad, establezca los propietarios posibles para ambos nodos. Establezca los recursos dependientes en none (ninguno).
6. Cree una instancia de recurso para la dirección IP.

   Cree un recurso de dirección IP (tipo de recurso *dirección IP*). Esta dirección debe ser una dirección IP no utilizada que los clientes y otros gestores de colas utilizarán para conectarse con el gestor de colas *virtual*. Esta dirección IP no es la dirección normal (estática) de cada nodo; es una dirección adicional que *flota* entre ellos. Aunque MSCS maneja el direccionamiento de esta dirección, **no** comprueba si se ha llegado a la dirección.
7. Cree una instancia de recurso para el gestor de colas.

Cree un recurso de tipo *IBM MQ MSCS*. El asistente le solicita varios elementos, entre los que se incluyen los siguientes:

- `Name` elija un nombre que facilite la identificación del gestor de colas para el que se encuentra.
- `Add to group` utilizar el grupo que ha creado
- `Run in a separate Resource Monitor` para un mejor aislamiento
- `Possible owners` establecer ambos nodos
- `Dependencies` añadir la unidad y la dirección IP para este gestor de colas.

  Aviso: No añadir estas dependencias significa que IBM MQ intenta grabar el estado del gestor de colas en el disco en clúster incorrecto durante las sustituciones por anomalía. Dado que puede haber muchos procesos que estén intentando grabar en este disco simultáneamente, algunos procesos de IBM MQ podrían bloquearse para impedir su ejecución.
- `Parameters` como sigue:
  - `QueueManagerName` (obligatorio); el nombre del gestor de colas que este recurso va a controlar. Este gestor de colas debe existir en el sistema local.
  - `PostOnlineCommand` (opcional); puede especificar un programa para ejecutarlo cuando el estado del recurso de gestor de colas pase de fuera de línea a en línea. Si desea ver información más detallada, consulte "PostOnlineCommand y PreOfflineCommand en MSCS" en la página 528.
  - `PreOfflineCommand` (opcional); puede especificar un programa para ejecutarlo cuando el estado del recurso de gestor de colas pase de en línea a fuera de línea. Si desea ver información más detallada, consulte "PostOnlineCommand y PreOfflineCommand en MSCS" en la página 528.

    Nota: El intervalo de sondeo *looksAlive* se establece en el valor predeterminado de 5000 ms. El intervalo de sondeo *isAlive* se establece en el valor predeterminado de 30000 ms. Estos valores predeterminados sólo se pueden modificar una vez que se ha completado la definición del recurso. Para obtener más información, consulte "Sondeo de looksAlive y isAlive en MSCS" en la página 525.

8. Opcionalmente, establezca un nodo preferido (pero tome nota de los comentarios en "Utilización de nodos preferidos en MSCS" en la página 529)
9. La *Política de sustitución por anomalía* (tal como se define en las propiedades del grupo) se establece de forma predeterminada en valores sensatos pero puede ajustar los umbrales y los períodos que controlan la *Sustitución por anomalía de recurso* y la *Sustitución por anomalía de grupo* para que coincida con la carga que se coloca en el gestor de colas.
10. Compruebe el gestor de colas pasándolo a en línea en el Administrador de clústeres de MSCS y sometiéndolo a una prueba de carga de trabajo. Si está experimentando con un gestor de colas de prueba, utilice IBM MQ Explorer. Por ejemplo:
    a. Pulse con el botón derecho del ratón en el nodo de árbol `Colas` y, a continuación, seleccione **Nueva** > **Cola local ...**, y asigne un nombre a la cola.
    b. Pulse **Finalizar**. Se crea la cola y aparece en la vista de contenido.
    c. Pulse el botón derecho del ratón en la cola y seleccione **Transferir mensaje de prueba...**. Se muestra el panel **Transferir mensaje de prueba**.
    d. Escriba el texto del mensaje, luego pulse **Transferir mensaje de prueba** y cierre el panel.
    e. Pulse el botón derecho del ratón en la cola y seleccione **Examinar mensajes...**. Se muestra el panel **Examinador de mensajes**.
    f. Asegúrese de que el mensaje está en la cola y a continuación pulse **Cerrar**. Se cierra el panel **Examinador de mensajes**.
    g. Pulse el botón derecho del ratón en la cola y seleccione **Borrar mensajes...**. Se borran los mensajes de la cola.

h. Pulse el botón derecho del ratón en la cola y seleccione **Suprimir...**. Aparece un panel de confirmación, pulse **Aceptar**. Se suprime la cola.

11. Compruebe que el gestor de colas se pueda pasar a fuera de línea y volver a pasar a en línea mediante el Administrador de clústeres de MSCS.
12. Simule una sustitución por anomalía.

    En el Administrador de clústeres de MSCS, pulse con el botón derecho del ratón en el grupo que contiene el gestor de colas y seleccione `Move Group`. Esto puede tardar algunos minutos. (Si en otro momento desea mover rápidamente un gestor de colas a otro nodo, siga el procedimiento de "Mover un gestor de colas al almacenamiento de MSCS" en la página 517.) También puede pulsar con el botón derecho del ratón y seleccionar `Initiate Failure` ; la acción (reinicio local o migración tras error) depende del estado actual y de los valores de configuración.

Windows *Sondeo de looksAlive y isAlive en MSCS*

looksAlive y *isAlive* son intervalos en los que Microsoft Cluster Service (MSCS) vuelve a llamar al código de biblioteca proporcionado por los tipos de recurso y solicita que el recurso realice comprobaciones para determinar el estado de trabajo de sí mismo. Esto determina finalmente si MSCS intenta una sustitución por anomalía del recurso.

Nota: A partir de Windows Server 2016, el nuevo nombre para Microsoft Cluster Service (MSCS) es Windows Server Failover Clustering (WSFC).

Cada vez que transcurre el intervalo *looksAlive* (valor predeterminado de 5000 ms), se llama al recurso de gestor de colas para que realice su propia comprobación para determinar si su estado es satisfactorio.

Cada vez que transcurre el intervalo *isAlive* (valor predeterminado de 30000 ms), se realiza otra llamada al recurso de gestor de colas para que realice otra comprobación para determinar si el recurso está funcionando correctamente. Esto permite dos niveles de comprobación de tipo de recurso.

1. Una comprobación de estado *looksAlive* para determinar si el recurso parece estar en funcionamiento.
2. Una comprobación *isAlive* más importante que determina si el recurso de gestor de colas está activo.

Si se determina que el recurso del gestor de colas no está activo, MSCS, basándose en otras opciones avanzadas de MSCS, desencadena una migración tras error para el recurso y los recursos dependientes asociados a otro nodo del clúster. Para obtener más información, consulte la documentación de MSCS.

Windows *Eliminar un gestor de colas del control de MSCS*

Puede eliminar gestores de colas del control de Microsoft Cluster Service (MSCS) y devolverlos a la administración manual.

Nota: A partir de Windows Server 2016, el nuevo nombre para Microsoft Cluster Service (MSCS) es Windows Server Failover Clustering (WSFC).

No es necesario eliminar gestores de colas del control de MSCS para las operaciones de mantenimiento. Puede hacerlo poniendo un gestor de colas fuera de línea temporalmente, mediante el Administrador de clústeres de MSCS. Eliminar un gestor de colas del control de MSCS es un cambio más permanente y solamente deberá hacerlo si decide que ya no desea que MSCS tenga ningún control adicional sobre el gestor de colas.

Si el gestor de colas que se está eliminando utiliza conexiones TLS, debe modificar el atributo de gestor de colas, SSLKEYR, utilizando IBM MQ Explorer o el mandato MQSC `ALTER QMGR`, para apuntar al archivo del repositorio de claves TLS en el directorio local.

Realice el siguiente procedimiento:

1. Ponga el recurso del gestor de colas fuera de línea utilizando el administrador de clústeres MSCS, tal como se describe en la "Poner un gestor de colas fuera de línea desde MSCS" en la página 526
2. Destruya la instancia del recurso. Esta acción no destruye el gestor de colas.
3. Opcionalmente, vuelva a migrar los archivos del gestor de colas de las unidades compartidas a las unidades locales. Para hacerlo, consulte "Devolver un gestor de colas desde el almacenamiento de MSCS" en la página 526.

4. Pruebe el gestor de colas.

Poner un gestor de colas fuera de línea desde MSCS

Para poner un gestor de colas fuera de línea desde MSCS, realice los pasos siguientes:

1. Inicie el Administrador de clústeres de MSCS.
2. Abra una conexión con el clúster.
3. Seleccione `Groupso Role` si está utilizando Windows 2012 y abra el grupo que contiene el gestor de colas que se va a mover.
4. Seleccione los recursos del gestor de colas.
5. Púlselo con el botón derecho del ratón y seleccione `Offline`.
6. Espere a que finalice.

Devolver un gestor de colas desde el almacenamiento de MSCS

Este procedimiento configura el gestor de colas para que vuelva a estar en la unidad local de su sistema, es decir, para que se convierta en un gestor de colas *normal* de IBM MQ. Para realizarlo, traslade los archivos de anotaciones y los archivos de datos de los discos compartidos. Por ejemplo, el gestor de colas existente puede tener vías de acceso como `E:\WebSphere MQ\log\`*QMname* y `E:\WebSphere MQ\qmgrs\`*QMname*. No intente mover los archivos a mano; utilice el programa de utilidad **hamvmqm** que se proporciona como parte del soporte MSCS de IBM MQ:

1. Efectúe una copia de seguridad de soportes completa de los archivos de colas y de los archivos de registro y guarde la copia de seguridad en un lugar seguro (consulte "Archivos de anotaciones del gestor de colas" en la página 528 si desea saber por qué esto es importante).
2. Decida qué unidad local debe utilizar para asegurarse de que tenga capacidad suficiente para almacenar los archivos de anotaciones y los archivos (de colas) de datos del gestor de colas.
3. Asegúrese de que el disco compartido donde residen actualmente los archivos está en línea en el nodo del clúster al que se trasladarán los archivos de anotaciones y de datos.
4. Ejecute el programa de utilidad para mover el gestor de colas como se indica a continuación:

   ```
   hamvmqm /m qmname /dd " c:\
   IBM MQ " /ld "c:\
   IBM MQ \log"
   ```

   sustituyendo *qmname* por el nombre de su gestor de colas, *c* por su letra de unidad de disco local e *IBM MQ* por el directorio que ha elegido (los directorios se crean si aún no existen).
5. Pruebe el gestor de colas para asegurarse de que funciona (tal como se describe en "Mover un gestor de colas al almacenamiento de MSCS" en la página 517).

Windows *Consejos y sugerencias sobre la utilización de MSCS*

Esta sección contiene información general para ayudarle a utilizar el soporte de IBM MQ para Microsoft Cluster Service (MSCS) de forma eficaz.

Nota: A partir de Windows Server 2016, el nuevo nombre para Microsoft Cluster Service (MSCS) es Windows Server Failover Clustering (WSFC).

¿Cuánto tiempo se tarda en pasar un gestor de colas de una máquina a otra durante una sustitución por anomalía? Esto dependerá mucho del volumen de la carga de trabajo que hay en el gestor de colas y de la combinación de tráfico, por ejemplo, qué cantidad del mismo es persistente, si está dentro del punto de sincronización y qué cantidad se ha confirmado antes de la anomalía. Las pruebas de IBM han dado tiempos de sustitución por anomalía y de recuperación de aproximadamente un minuto. Las pruebas se realizaron en un gestor de colas con una carga de trabajo considerable y el tiempo real puede variar mucho dependiendo de la carga.

Windows *Verificación del funcionamiento de MSCS*

Siga estos pasos para asegurarse de que tiene un clúster de MSCS en ejecución.

Las descripciones de tareas que comienzan en el apartado "Creación de un gestor de colas para utilizarlo con MSCS" en la página 516 dan por supuesto que tiene un clúster de MSCS en ejecución en el que puede crear, migrar y destruir recursos. Si desea asegurarse de que tiene este tipo de clúster:

1. Mediante el Administrador de clústeres de MSCS, cree un grupo.
2. Dentro de ese grupo, cree una instancia de un recurso de aplicación genérico, especificando el reloj del sistema (nombre de vía de acceso `C:\winnt\system32\clock.exe` y directorio de trabajo de `C:\`).
3. Asegúrese de que puede poner el recurso en línea, de que puede trasladar el grupo que lo contiene al otro nodo y de que puede poner el recurso fuera de línea.

Windows *Inicio manual y MSCS*

Para un gestor de colas gestionado por MSCS, debe establecer el atributo de inicio en manual. Esto asegura que el soporte MSCS de IBM MQ pueda reiniciar el servicio MQSeries sin iniciar inmediatamente el gestor de colas.

El soporte MSCS de IBM MQ tiene que poder reiniciar el servicio para que éste pueda realizar las tareas de supervisión y control, pero debe, a su vez, seguir controlando qué gestores de colas se están ejecutando y en qué máquinas. Consulte "Mover un gestor de colas al almacenamiento de MSCS" en la página 517 para obtener más información.

Windows *MSCS y los gestores de colas*

Consideraciones relativas a los gestores de colas cuando utilizan MSCS.

Creación de un gestor de colas coincidente en el otro nodo

Para que la agrupación en clúster funcione con IBM MQ, necesita un gestor de colas idéntico en el nodo B para cada uno en el nodo A. Sin embargo, no es necesario que cree explícitamente el segundo. Puede crear o preparar un gestor de colas en un nodo, pasarlo al otro nodo como se describe en "Mover un gestor de colas al almacenamiento de MSCS" en la página 517, y se duplicará totalmente en ese nodo.

Gestores de colas predeterminados

No utilice un gestor de colas predeterminado bajo el control de MSCS. Un gestor de colas no tiene una propiedad que lo convierte en el gestor predeterminado; IBM MQ mantiene su propio registro separado. Si durante una sustitución por anomalía traslada al otro sistema un gestor de colas que se ha establecido como el gestor de colas predeterminado, éste no se convierte en el gestor de colas predeterminado en el otro sistema. Haga que todas sus aplicaciones hagan referencia a gestores de cola específicos por su nombre.

Supresión de un gestor de colas

Cuando un gestor de colas ha cambiado de un nodo a otro, su información detallada existe en el registro de ambos sistemas. Cuando desee suprimirlo, hágalo normalmente en un sistema y luego ejecute el programa de utilidad que se describe en "Soporte para programas de utilidad MSCS" en la página 529 para limpiar el registro en el otro sistema.

Soporte para los gestores de colas existentes

Puede poner un gestor de colas existente bajo el control de MSCS, siempre que pueda colocar los archivos de anotaciones y los archivos de colas del gestor de colas en un disco que esté en el bus SCSI compartido entre dos máquinas (vea la Figura 71 en la página 514). Durante el breve período de tiempo que dura la creación del recurso MSCS, el gestor de colas deberá estar fuera de línea.

Si desea crear un nuevo gestor de colas, créelo independientemente de MSCS, pruébelo y luego póngalo bajo el control de MSCS. Consulte:

- “Creación de un gestor de colas para utilizarlo con MSCS” en la página 516
- “Mover un gestor de colas al almacenamiento de MSCS” en la página 517
- “Poner un gestor de colas bajo control de MSCS” en la página 518

Indicar a MSCS qué gestores de colas debe gestionar

Puede seleccionar qué gestores de colas se ponen bajo el control de MSCS utilizando el Administrador de clústeres de MSCS para crear una instancia de recurso para cada gestor de colas de este tipo. Este proceso le presenta una lista de recursos en la que puede seleccionar el gestor de colas que desea que gestione dicha instancia.

Archivos de anotaciones del gestor de colas

Cuando pasa un gestor de colas al almacenamiento de MSCS, traslada sus archivos de anotaciones y de datos a un disco compartido (para ver un ejemplo, consulte “Mover un gestor de colas al almacenamiento de MSCS” en la página 517).

Antes de realizar el traslado, es recomendable concluir el gestor de colas de forma ordenada y efectuar una copia de seguridad completa de los archivos de datos y de los archivos de anotaciones.

Varios gestores de colas

El soporte MSCS de IBM MQ le permite ejecutar varios gestores de colas en cada máquina y colocar gestores de colas individuales bajo control de MSCS.

Windows *Utilice siempre MSCS para gestionar clústeres*

No intente realizar operaciones de inicio y detención directamente en ningún gestor de colas bajo el control de MSCS, ya sea mediante los mandatos de control o IBM MQ Explorer. En su lugar, utilice el Administrador de clústeres de MSCS para colocar el gestor de colas en línea o fuera de línea.

El Administrador de clústeres de MSCS se utiliza para impedir en parte la posible confusión que puede ocasionar que MSCS informe de que el gestor de colas está fuera de línea, cuando de hecho ha iniciado el gestor de colas fuera del control de MSCS. Pero lo que resulta más grave es detener un gestor de colas sin utilizar MSCS, ya que MSCS detecta está operación como una anomalía e inicia la sustitución por anomalía en el otro nodo.

Windows *Trabajar en modalidad Activa/Activa en MSCS*

Los dos sistemas del clúster de MSCS pueden ejecutar gestores de colas en modalidad Activa/Activa. No es necesario que tenga una máquina completamente desocupada que actúe como máquina de reserva (pero, si lo desea, puede hacerlo en la modalidad Activa/Pasiva).

Si piensa utilizar ambas máquinas para ejecutar la carga de trabajo, proporcione a cada una la capacidad suficiente (memoria de procesador, almacenamiento secundario) para ejecutar toda la carga de trabajo del clúster con un nivel de rendimiento satisfactorio.

Nota: Si utiliza MSCS junto con Microsoft Transaction Server (COM+), **no puede** utilizar la modalidad Activa/Activa. Esto se debe a que, para utilizar IBM MQ con MSCS y COM+:

- Los componentes de la aplicación que utilizan el soporte COM+ de IBM MQ deben ejecutarse en el mismo sistema que el Coordinador de transacciones distribuidas (DTC), un componente de COM+.
- El gestor de colas también debe ejecutarse en el mismo sistema.
- El DTC debe configurarse como un recurso de MSCS y, por lo tanto, sólo puede ejecutarse en uno de los sistemas del clúster a la vez.

Windows *PostOnlineCommand y PreOfflineCommand en MSCS*

Utilice estos mandatos para integrar el soporte MSCS de IBM MQ con otros sistemas. Puede utilizarlos para emitir mandatos de IBM MQ, con algunas restricciones.

Especifique estos mandatos en los parámetros para un recurso de tipo `IBM MQ MSCS`. Puede utilizarlos para integrar el soporte MSCS de IBM MQ con otros sistemas o procedimientos. Por ejemplo, puede especificar el nombre de un programa que envíe un mensaje de correo, active un buscapersonas o genere algún otro tipo de alerta que capturará otro sistema de supervisión.

PostOnlineCommand se invoca cuando el recurso pasa de estar fuera de línea a estar en línea y PreOfflineCommand se invoca para un cambio de en línea a fuera de línea. Cuando se invocan, estos mandatos se ejecutan, de forma predeterminada, desde el directorio del sistema Windows. Puesto que IBM MQ utiliza un proceso de supervisión de recursos de 32 bits, en sistemas Windows de 64 bits, este es el directorio `\Windows\SysWOW64` en lugar del directorio `\Windows\system32`. Para obtener más información, consulte la documentación de Microsoft sobre la redirección de archivos en un entorno Windows x64. Ambos mandatos se ejecutan bajo la cuenta de usuario que se utiliza para ejecutar el servicio de clúster MSCS y se invocan de forma asíncrona; el soporte MSCS de IBM MQ no espera a que se completen para continuar. De este modo, se elimina el riesgo de que bloqueen o retrasen otras operaciones del clúster.

También puede utilizar estos mandatos para emitir mandatos de IBM MQ, por ejemplo, para reiniciar canales peticionarios. No obstante, los mandatos se ejecutan en el momento en que el estado del gestor de colas cambia, por lo tanto, no se han diseñado para realizar funciones de larga ejecución y no se debe presuponer el estado actual del gestor de colas; es muy probable que, inmediatamente después de que el gestor de colas pasa a estar en línea, un administrador haya emitido un mandato fuera de línea.

Si desea ejecutar programas que dependen del estado del gestor de colas, considere la posibilidad de crear instancias del tipo de recurso `MSCS Generic Application` , colocarlas en el mismo grupo MSCS que el recurso del gestor de colas y hacerlas dependientes del recurso del gestor de colas.

Windows *Utilización de nodos preferidos en MSCS*

Puede ser útil cuando se utiliza la modalidad Activa/Activa en MSCS para configurar un *nodo preferido* para cada gestor de colas. Sin embargo, en general es mejor no establecer un nodo preferido y basarse en una sustitución por anomalía manual.

A diferencia de otros recursos que relativamente no tienen estado, durante el proceso de sustitución por anomalía, un gestor de colas puede tardar algún tiempo en pasar de un nodo a otro. Para evitar las interrupciones innecesarias, compruebe el nodo recuperado antes de devolver al mismo el gestor de colas que se ha sustituido por anomalía. Esto impide el uso del valor de restablecimiento de `immediate` . Puede configurar la sustitución por anomalía de modo que se produzca a determinadas horas del día.

Probablemente, la forma más segura sea volver a colocar el gestor de colas en el nodo necesario manualmente cuando esté seguro de que el nodo se ha recuperado totalmente. Esto impide el uso de la opción `preferred node` .

Windows *Errores COM+ al instalar en MSCS*

Cuando instala IBM MQ en un clúster de MSCS recién instalado, puede que encuentre un error con Origen COM+ e ID de suceso 4691 en el registro de sucesos de aplicación.

Esto significa que está intentando ejecutar IBM MQ en un entorno Microsoft Cluster Server (MSCS) cuando el Coordinador de transacciones distribuidas de Microsoft (MSDTC) no se ha configurado para ejecutarse en un entorno de este tipo. Para obtener información sobre la configuración de MSDTC en un entorno en clúster, consulte la documentación de Microsoft.

Windows *Soporte para programas de utilidad MSCS*

Una lista de los programas de utilidad de soporte de IBM MQ para Microsoft Cluster Service (MSCS) que puede ejecutar en un indicador de mandatos.

Nota: A partir de Windows Server 2016, el nuevo nombre para Microsoft Cluster Service (MSCS) es Windows Server Failover Clustering (WSFC).

El soporte de IBM MQ para MSCS incluye los siguientes programas de utilidad:

Registrar/desregistrar el tipo de recurso

`haregtyp.exe`

Después de *anular el registro* del tipo de recurso MSCS de IBM MQ ya no puede crear ningún recurso de ese tipo. MSCS no le permite anular el registro de un tipo de recurso si todavía tiene instancias de dicho tipo en el clúster:

1. Con el Administrador de clústeres de MSCS, detenga los gestores de colas que estén ejecutándose bajo el control de MSCS poniéndolos fuera de línea como se describe en "Poner un gestor de colas fuera de línea desde MSCS" en la página 526.
2. Mediante el Administrador de clústeres de MSCS, suprima las instancias de recursos.
3. En un indicador de mandatos, anule el registro del tipo de recurso entrando el mandato siguiente:

```
haregtyp /u
```

Si desea *registrar* el tipo (o volver a registrarlo posteriormente), escriba el mandato siguiente en un indicador de mandatos:

```
haregtyp /r
```

Después de registrar satisfactoriamente las bibliotecas MSCS, debe reiniciar el sistema si no lo ha hecho desde la instalación de IBM MQ.

Mover un gestor de colas al almacenamiento de MSCS

`hamvmqm.exe`

Consulte "Mover un gestor de colas al almacenamiento de MSCS" en la página 517.

Suprimir un gestor de colas de un nodo

`hadltmqm.exe`

Considere el caso en el que ha tenido un gestor de colas en el clúster, ha sido trasladado de un nodo a otro y ahora desea eliminarlo. Utilice IBM MQ Explorer para suprimirlo en el nodo donde se encuentra actualmente. Las entradas de registro del gestor de colas siguen existiendo en el otro sistema. Para suprimirlas, escriba el mandato siguiente en un indicador de mandatos de dicho sistema:

```
hadltmqm /m qmname
```

donde `qmname` es el nombre del gestor de colas que se debe eliminar.

Comprobar y guardar los detalles de configuración

`amqmsysn.exe`

Este programa de utilidad presenta un diálogo que muestra todos los detalles de la configuración del soporte MSCS de IBM MQ, como los que se le podrían solicitar si llamase al centro de soporte de IBM. Tiene la opción de guardar la información detallada en un archivo.

Multi Gestores de colas multiinstancia

Los gestores de colas multiinstancia son instancias del mismo gestor de cola configuradas en diferentes servidores. Una instancia del gestor de colas se define como la instancia activa y otra instancia se define como la instancia en espera. Si la instancia activa falla, el gestor de colas multiinstancia se reinicia automáticamente en el servidor en espera.

Ejemplo de configuración de gestor de colas multiinstancia

La Figura 72 en la página 531 muestra un ejemplo de una configuración multiinstancia para el gestor de colas QM1. IBM MQ está instalado en dos servidores, uno de los cuales es de repuesto. Se ha creado un gestor de colas, QM1. Una instancia de QM1 está activa y se está ejecutando en un servidor. La otra instancia de QM1 se está ejecutando en espera en el otro servidor, no está realizando ningún proceso de forma activa, pero está preparada para sustituir a la activa de QM1 en caso de que falle. (Sólo puede haber una instancia activa y una instancia en espera del gestor de colas en una configuración de varias instancias.)

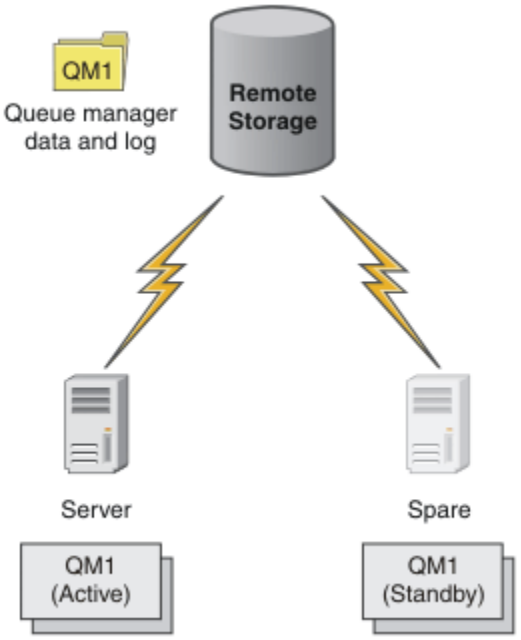


Figura 72. Gestor de colas multiinstancia

Cuando intente utilizar un gestor de colas como gestor de colas multiinstancia, cree un gestor de colas único en uno de los servidores mediante el mandato **crtmqm**, colocando sus datos y registros en un almacenamiento en red compartido. En el otro servidor, en vez de crear un gestor de colas de nuevo, utilice el mandato **addmqinf** para crear una referencia a los registros y datos del gestor de colas en el almacenamiento en red.

Ahora puede ejecutar el gestor de colas desde cualquiera de los servidores. Ambos servidores hacen referencia a los mismos registros y datos del gestor de colas; sólo hay un gestor de colas y está activo sólo en un servidor al mismo tiempo.

El gestor de colas puede ejecutarse como un gestor de colas de una sola instancia o como un gestor de colas multiinstancia. En ambos casos sólo se ejecuta una instancia del gestor de colas al procesar solicitudes. La diferencia estriba en que al ejecutarse como un gestor de colas multiinstancia, el servidor que no está ejecutando la instancia activa del gestor de colas ejecuta también una instancia en espera, que está lista para sustituir automáticamente a la activa si el servidor activo falla.

El único control que se puede tener sobre qué instancia se convierte en activa primero es el orden en el que se inicia el gestor de colas en los dos servidores. La primera instancia que consigue leer/grabar bloqueos en los datos del gestor de colas pasa a ser la instancia activa.

Se puede alternar la instancia activa entre uno y otro servidor, una vez que se haya iniciado, deteniendo la instancia activa mediante la opción de cambio para transferir el control a la que está en espera.

La instancia activa de QM1 tiene acceso exclusivo a las carpetas de registros y datos del gestor de colas compartido cuando se está ejecutando. La instancia en espera de QM1 detecta cuando existe alguna anomalía en la instancia activa y pasa a ser ella la activa. Retoma los registros y datos de QM1 en el estado que las ha dejado la instancia activa y acepta las reconexiones desde los clientes y canales.

La instancia activa puede fallar por diversas razones que dan lugar a que la instancia en espera la sustituya:

- Por una anomalía del servidor que aloja la instancia del gestor de colas activo.
- Anomalía de la conectividad entre el servidor que aloja la instancia activa del gestor de colas y el sistema de archivos.
- Falta de respuesta de los procesos de gestor de colas, detectada por IBM MQ, que a continuación concluye el gestor de colas.

Se puede añadir información de configuración del gestor de colas a varios servidores y elegir entre dos servidores para que ejecuten la instancia activa y pasiva. Hay un límite de un total de dos instancias. No puede tener dos instancias en espera y una instancia activa.

Componentes adicionales necesarios para crear una solución de alta disponibilidad

Un gestor de colas multiinstancia es una parte de una solución de alta disponibilidad. Se necesitan otros componentes adicionales para crear una solución de alta disponibilidad útil.

- Reconexión de cliente y canal para transferir conexiones de IBM MQ al sistema que toma el control ejecutando la instancia activa de gestor de colas.
- Un sistema de archivos de red (NFS) compartido de alto rendimiento que gestione bloqueos correctamente y proporcione protección frente a anomalías de soporte y de servidor de archivos.

  Importante: Debe detener todas las instancias de gestor de colas multiinstancia que estén ejecutándose en el entorno antes de poder realizar el mantenimiento en la unidad NFS. Asegúrese de que tiene copias de seguridad de configuración de gestor de colas para recuperar, en el caso de una anomalía de NFS.
- Redes y fuentes de alimentación resilientes para eliminar puntos únicos de anomalía en la infraestructura básica.
- Aplicaciones que toleren anomalías. En particular, es necesario prestar mucha atención al comportamiento de las aplicaciones transaccionales y a las aplicaciones que examinan las colas de IBM MQ.
- Supervisión y gestión de instancias activas y en espera para asegurarse de que se están ejecutando y para reiniciar las instancias activas que hayan fallado. Aunque los gestores de colas multiinstancia se reinician automáticamente, tiene que asegurarse de que las instancias en espera están ejecutándose, listas para tomar el control, y que las instancias con error vuelven a ponerse en línea como nuevas instancias en espera.

Los IBM MQ MQI clients y los canales se reconectan automáticamente al gestor de colas en espera cuando pasa a estar activo. Puede encontrar más información sobre la reconexión y los otros componentes en una solución de alta disponibilidad en los temas relacionados. La reconexión automática de cliente no está soportada en IBM MQ classes for Java.

Plataformas soportadas

Puede crear un gestor de colas de varias instancias en cualquier sistema Multiplatforms.

La reconexión automática de cliente está soportada para los clientes MQI.

Crear un gestor de colas multiinstancia

Cree un gestor de colas multiinstancia, creando el gestor de colas en un servidor y configurando IBM MQ en otro servidor. Los gestores de colas multiinstancia comparten datos y registros del gestor de colas.

La mayor parte del esfuerzo necesario para crear un gestor de colas multiinstancia consiste en configurar los datos y los archivos de registro y de datos compartidos del gestor de colas. Debe crear directorios compartidos en almacenamiento en red y poner los directorios a disposición de otros servidores utilizando unidades compartidas de red. Estas tareas deben ser realizadas por un usuario con autoridad administrativa, por ejemplo el usuario *root* en sistemas AIX and Linux. Los pasos son los siguientes:

1. Crear los compartimientos para los datos y los archivos de registros.
2. Crear el gestor de colas en un servidor.
3. Ejecutar el mandato **dspmqinf** en el primer servidor para recopilar los datos de configuración del gestor de colas y copiarlos en el portapapeles.
4. Ejecutar el mandato **addmqinf** con los datos copiados para crear la configuración del gestor de colas en el segundo servidor.

No ejecute **crtmqm** para crear de nuevo el gestor de colas en el segundo servidor.

Control de acceso a archivos

Debe procurar que el usuario y el grupo `mqm` en todos los demás servidores tengan permiso para acceder a las unidades compartidas.

En sistemas AIX and Linux, los valores de `uid` y `gid` de `mqm` deben ser los mismos en todos los sistemas. Es posible que necesite editar `/etc/passwd` en cada sistema para establecer los valores de `uid` y `gid` para `mqm`, con un valor común y reiniciar el sistema.

En Microsoft Windows, el ID de usuario que está ejecutando los procesos del gestor de colas debe tener un permiso de control completo sobre los directorios que contienen los datos y archivos de registro del gestor de colas. Puede configurar el permiso de dos maneras:

1. Cree un gestor de colas con un grupo global como el principal de seguridad alternativo. Autorice el grupo global para que tenga acceso de control completo sobre los directorios que contienen archivos de datos y registros del gestor de colas; consulte “Proteger directorios y archivos de datos y registros compartidos del gestor de colas en Windows” en la página 561. Haga que el ID de usuario que ejecuta el gestor de colas sea miembro del grupo global. No puede hacer que el usuario local sea miembro de un grupo global; por consiguiente, los procesos del gestor de colas deben ejecutarse bajo un ID de usuario de dominio. El ID de usuario de dominio debe ser miembro del grupo local `mqm`. La tarea, “Creación de gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio en Windows” en la página 536, muestra cómo configurar un gestor de colas multiinstancia utilizando archivos protegidos de esta forma.
2. Cree un gestor de colas en el controlador de dominio, de modo que el grupo `mqm` local tenga ámbito de dominio, "local del dominio". Proteja el compartimiento de archivos con el `mqm` local del dominio y ejecute los procesos del gestor de colas en todas las instancias de un gestor de colas bajo el mismo grupo `mqm` local del dominio. La tarea, “Creación de un gestor de colas multiinstancia en controladores de dominio de Windows” en la página 551, muestra cómo configurar un gestor de colas multiinstancia utilizando archivos protegidos de esta forma.

Información de configuración

Configure todas las instancias del gestor de colas que necesite modificando la información de configuración de gestor de colas de IBM MQ sobre cada servidor. Cada servidor debe tener instalada la misma versión de IBM MQ en un nivel de arreglo compatible. Los mandatos **`dspmqinf`** y **`addmqinf`** ayudan a configurar las instancias del gestor de colas adicionales. De forma alternativa, puede editar los archivos `mqs.ini` y `qm.ini` directamente. Los temas, “Creación de un gestor de colas multiinstancia en Linux” en la página 574, “Creación de gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio en Windows” en la página 536 y“Creación de un gestor de colas multiinstancia en controladores de dominio de Windows” en la página 551 son ejemplos que muestra cómo configurar un gestor de colas multiinstancia.

En sistemas AIX, Linux, and Windows, se puede compartir un único archivo `mqs.ini` colocándolo en el compartimento de red y configurando la variable de entorno **`AMQ_MQS_INI_LOCATION`** para que apunte a él.

Restricciones

1. Configure varias instancias del mismo gestor de colas sólo en servidores que tengan el mismo sistema operativo, arquitectura y endianidad. Por ejemplo, ambas máquinas deben tener un tamaño de palabras de 32 bits o 64 bits.
2. Todas las instalaciones de IBM MQ debe tener el nivel de release 7.0.1 o superior.
3. Normalmente, las instalaciones de instancias activas y en espera se mantienen en el mismo nivel de mantenimiento. Consulte las instrucciones de mantenimiento de cada actualización para comprobar si debe actualizar todas las instalaciones a la vez.

   Tenga en cuenta que los niveles de mantenimiento para los gestores de colas activos y pasivos deben ser idénticos.

4. Comparta datos y registros del gestor de colas únicamente entre gestores de colas que están configurados con el mismo usuario, grupo y mecanismo de control de acceso de IBM MQ.

   IBM i Por ejemplo, la configuración de la unidad compartida de red en un servidor Linux podría contener datos y registros separados del gestor de colas para gestores de colas AIX and Linux, pero podría no contener los datos del gestor de colas utilizado por IBM i.

   IBM i Se pueden crear varias unidades compartidas en el mismo almacenamiento en red para IBM i y para sistemas AIX and Linux siempre que las unidades compartidas sean diferentes. Pueden asignar diferentes propietarios a diferentes unidades compartidas. La restricción es una consecuencia de los distintos nombres utilizados para los usuarios y grupos de IBM MQ entre AIX and Linux e IBM i. El hecho de que el usuario y el grupo puedan tener el mismo `uid` y `gid` no reduce la restricción.
5. En sistemas AIX and Linux, configure el sistema de archivos compartidos en almacenamiento en red con un montaje `hard`, interrumpible, en vez de `soft`. Un montaje interrumpible hard fuerza al gestor de colas a mantenerse hasta que queda interrumpido por una llamada del sistema. Los montajes soft no garantizan la consistencia de datos después de que un servidor falle.
6. Los directorios de datos y registros compartidos no pueden almacenarse en un sistema de archivos FAT o NFSv3. Para los gestores de colas multiinstancia en Windows, el almacenamiento en red debe ser accesible para el protocolo CIFS (Common Internet File System) utilizado por las redes Windows.
7. z/OS z/OS no da soporte a gestores de colas multiinstancia. Utilice grupos de compartición de colas.

   Los clientes reconectables funcionan con gestores de colas de z/OS.

Windows *Dominios de Windows y gestores de colas multiinstancia*

Un gestor de colas multiinstancia en Windows requiere que se compartan sus datos y registros. El compartimiento debe ser accesible para todas las instancias del gestor de colas que se ejecutan en diferentes servidores o estaciones de trabajo. Configure los gestores de colas y compártalos como parte de un dominio de Windows. El gestor de colas se puede ejecutar en una estación de trabajo o servidor de dominio o en el controlador de dominio.

Importante: De forma predeterminada, los sistemas que empiezan por Windows 10 versión 1607 y Windows Server 2016 son más restrictivos que las versiones anteriores de Windows.

Este cambio restringe los clientes que pueden realizar llamadas remotas al Gestor de cuentas de seguridad (SAM) y podría afectar a IBM MQ con gestores de colas que no se pueden iniciar. El acceso a SAM es fundamental para el funcionamiento de IBM MQ cuando IBM MQ se configura como una cuenta de dominio.

Antes de configurar un gestor de colas multiinstancia, lea “Proteger directorios y archivos de datos y registros del gestor de colas no compartidos en Windows” en la página 564 y “Proteger directorios y archivos de datos y registros compartidos del gestor de colas en Windows” en la página 561 para revisar cómo controlar el acceso a archivos de datos y de registro del gestor de colas. Los temas son educativos; si desea ir directamente a configurar directorios compartidos para un gestor de colas multiinstancia en un dominio de Windows, consulte “Creación de gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio en Windows” en la página 536.

Ejecutar un gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio

A partir de la IBM WebSphere MQ 7.1, los gestores de colas multiinstancia se ejecutan en una estación de trabajo o un servidor que es miembro de un dominio. Para ejecutar un gestor de colas multiinstancia en Windows, se requiere un controlador de dominio, un servidor de archivos y dos estaciones de trabajo o servidores que ejecuten el mismo gestor de colas conectado al mismo dominio.

El cambio que hace posible ejecutar un gestor de colas multiinstancia en cualquier servidor o estación de trabajo en un dominio, es que ahora puede crear un gestor de colas con un grupo de seguridad adicional. El grupo de seguridad adicional se transfiere en el mandato **`crtmqm`**, en el parámetro `-a`. Proteja los directorios que contienen los datos del gestor de colas y los registros con el grupo. El ID de usuario

que ejecute procesos del gestor de colas debe ser miembro de este grupo. Cuando el gestor de colas accede a los directorios, Windows comprueba los permisos que el ID de usuario tiene para acceder a los directorios. Si otorga al grupo y al ID de usuario ámbito de dominio, el ID de usuario que ejecute los procesos del gestor de colas tendrá las credenciales del grupo global. Cuando el gestor de colas se ejecuta en otro servidor, el ID de usuario que ejecute los procesos del gestor de colas tiene las mismas credenciales. El ID de usuario no tiene que ser el mismo. Debe ser miembro del grupo de seguridad alternativo, así como miembro del grupo local `mqm`.

Consulte “Creación de gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio en Windows” en la página 536 para obtener detalles sobre la creación de un gestor de colas de varias instancias.

Se requieren varios pasos para configurar el dominio y los servidores de dominio y estaciones de trabajo. Es preciso que comprenda cómo Windows autoriza el acceso mediante un gestor de colas a sus directorios de registros y datos. Si no está seguro de cómo se autorizan los procesos del gestor de colas para acceder a los archivos de registros y datos, lea el tema del apartado “Proteger directorios y archivos de datos y registros del gestor de colas no compartidos en Windows” en la página 564. El tema incluye dos tareas para ayudarle a comprender los pasos necesarios. La tareas son “Lectura y grabación de datos y archivos de registro autorizados por el grupo mqm local” en la página 566 y “Leer y grabar archivos de datos y de registro autorizados por un grupo de seguridad local alternativo” en la página 569. Otro tema, “Proteger directorios y archivos de datos y registros compartidos del gestor de colas en Windows” en la página 561, explica cómo proteger directorios compartidos que contengan archivos de datos y registros del gestor de datos con el grupo de seguridad alternativo. El tema incluya cuatro tareas para configurar un dominio de Windows, crear una unidad compartida de archivos, instalar IBM MQ for Windows y configurar un gestor de colas para utilizar la unidad compartida. Las tareas son las siguientes:

1. “Creación de un dominio de Active Directory y DNS en Windows” en la página 539.
2. “Instalar IBM MQ en un servidor o una estación de trabajo de un dominio de Windows” en la página 542.
3. “Creación de un directorio compartido para los archivos de datos y de registro del gestor de colas en Windows” en la página 545.
4. “Leer y grabar archivos de datos y de registro compartidos autorizados por un grupo de seguridad global alternativo” en la página 548.

A continuación, puede realizar la tarea,“Creación de gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio en Windows” en la página 536, utilizando el dominio. Efectúe estas tareas para explorar la configuración de un gestor de colas multiinstancia antes de transferir el conocimiento a un dominio de producción.

Ejecutar un gestor de colas multiinstancia en controladores de dominio

Los datos del gestor de colas podían protegerse con el grupo `mqm` del dominio. Tal como se explica en el tema “Proteger directorios y archivos de datos y registros compartidos del gestor de colas en Windows” en la página 561, no puede compartir directorios protegidos con un grupo `mqm` local en estaciones de trabajo o servidores. No obstante, en controladores de dominio todos los grupos y principales tienen ámbito de dominio. Si instala IBM MQ for Windows en un controlador de dominio, los archivos de datos y registros del gestor de colas están protegidos con el grupo `mqm` del dominio, que puede compartirse. Siga los pasos de la tarea “Creación de un gestor de colas multiinstancia en controladores de dominio de Windows” en la página 551 para configurar un gestor de colas multiinstancia en controladores de dominio.

Información relacionada

Gestión de autorización y control de acceso

Cómo utilizar nodos de clúster de Windows Server como controladores de dominio

Windows *Creación de gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio en Windows*

Un ejemplo muestra cómo se configura un gestor de colas multiinstancia en Windows en una estación de trabajo o un servidor que forme parte de un dominio de Windows. El servidor no tiene que ser un controlador de dominio. La configuración muestra los conceptos implicados en vez de realizarse a una escala de producción. El ejemplo se basa en Windows Server 2008. Los pasos pueden ser diferentes en otras versiones de Windows Server.

En una configuración a escala de producción, puede que deba ajustar la configuración a un dominio existente. Por ejemplo, podría definir diferentes grupos de dominio para autorizar diferentes unidades compartidas y para agrupar los ID de usuario que ejecutan gestores de colas.

La configuración del ejemplo consta de tres servidores:

sun
: Un controlador de dominio Windows Server 2008. Es propietario del dominio *wmq.example.com* que contiene *Sun*, *mars* y *venus*. Para ilustrar esto, también se utiliza el servidor de archivos.

mars
: Un Windows Server 2008 utilizado como primer servidor de IBM MQ. Contiene una instancia del gestor de colas de varias instancias denominado *QMGR*.

venus
: Un Windows Server 2008 utilizado como segundo servidor de IBM MQ. Contiene la segunda instancia del gestor de colas de varias instancias denominado *QMGR*.

Sustituya los nombres en cursiva del ejemplo, por los nombres que desee.

Antes de empezar

En Windows, no es necesario verificar el sistema de archivos en el que tiene pensado guardar los archivos de datos y registros del gestor de colas. El procedimiento de comprobación, Verificación del comportamiento del sistema de archivos compartidos, es aplicable a AIX and Linux. En Windows, las comprobaciones siempre son satisfactorias.

Efectúe los pasos de las tareas siguientes. Las tareas crean el controlador de dominio y el dominio, instalan IBM MQ for Windows en un servidor y crean la compartición de archivos para los datos y los archivos de registro. Si está configurando un controlador de dominio existente, puede resultar útil intentar los pasos en un nuevo Windows Server 2008. Puede adaptar los pasos a su dominio.

1. “Creación de un dominio de Active Directory y DNS en Windows” en la página 539.
2. “Instalar IBM MQ en un servidor o una estación de trabajo de un dominio de Windows” en la página 542.
3. “Creación de un directorio compartido para los archivos de datos y de registro del gestor de colas en Windows” en la página 545.
4. “Leer y grabar archivos de datos y de registro compartidos autorizados por un grupo de seguridad global alternativo” en la página 548.

Acerca de esta tarea

Esta tarea pertenece a una secuencia de tareas para configurar un controlador de dominio y dos servidores en el dominio con el fin de ejecutar instancias de un gestor de colas. En esta tarea se configura un segundo servidor, *venus*, para ejecutar otra instancia del gestor de colas *QMGR*. Siga los pasos que se indican en esta tarea para crear la segunda instancia del gestor de colas, *QMGR* y comprobar si funciona.

Esta tarea es separada de las cuatro tareas mencionadas en el apartado anterior. Contiene los pasos que convierten un gestor de colas de una sola instancia en un gestor de colas multiinstancia. Todos los demás pasos son comunes a los gestores de colas de una sola instancia o multiinstancia.

Procedimiento

1. Configure un segundo servidor para ejecutar IBM MQ for Windows.
   a) Realice los pasos de la tarea “Instalar IBM MQ en un servidor o una estación de trabajo de un dominio de Windows” en la página 542 para crear un segundo servidor de dominio. En esta secuencia de tareas se llama al segundo servidor *venus*.

      Consejo: Cree la segunda instalación utilizando los mismos valores predeterminados de instalación para IBM MQ en cada uno de los dos servidores. Si los valores predeterminados difieren, es posible que tenga que adaptar las variables `Prefix` e `InstallationName` en la stanza *QMGR* **QueueManager** del archivo de configuración de IBM MQ `mqs.ini`. Las variables hacen referencia a vías de acceso que pueden ser diferentes para cada instalación y gestor de colas en cada servidor. Si las vías de acceso siguen siendo las mismas en cada servidor, es más sencillo configurar un gestor de colas multiinstancia.
2. Cree una segunda instancia de *QMGR* en *venus*.
   a) Si *QMGR* en *mars* no existe, realice la tarea “Leer y grabar archivos de datos y de registro compartidos autorizados por un grupo de seguridad global alternativo” en la página 548, para crearla
   b) Compruebe que los valores de los parámetros `Prefix` e `InstallationName` son correctos para *venus*.

      En *mars*, ejecute el mandato **dspmqinf**:

      ```
      dspmqinf QMGR
      ```

      La respuesta del sistema:

      ```
      QueueManager:
      Name=QMGR
      Directory=QMGR
      Prefix= C:\ProgramData \IBM \MQ
      DataPath=\\sun\wmq\data\QMGR
      InstallationName=Installation1
      ```

   c) Copie el formato legible por máquina de la stanza **QueueManager** en el portapapeles.

      En *mars* , vuelva a ejecutar el mandato **dspmqinf** , con el parámetro `-o command` .

      ```
      dspmqinf -o command QMGR
      ```

      La respuesta del sistema:

      ```
      addmqinf -s QueueManager -v Name=QMGR
      -v Directory=QMGR -v Prefix=" C:\ProgramData \IBM \MQ"
      -v DataPath=\\sun\wmq\data\QMGR
      ```

   d) En *venus*, ejecute el mandato **addmqinf** del portapapeles para crear una instancia del gestor de colas en *venus*.

      Ajuste el mandato si es necesario, para acomodar las diferencias en los parámetros `Prefix` o `InstallationName`.

      ```
      addmqinf -s QueueManager -v Name=QMGR
      -v Directory=QMGR -v Prefix=" C:\ProgramData \IBM \MQ"
      -v DataPath=\\sun\wmq\data\QMGR
      ```

      ```
      IBM MQ configuration information added.
      ```

3. Inicie el gestor de colas *QMGR* en *venus*, permitiendo instancias en espera.
   a) Compruebe si *QMGR* en *mars* se ha detenido.

      En *mars*, ejecute el mandato **dspmq**:

      ```
      dspmq -m QMGR
      ```

      La respuesta del sistema depende de cómo se haya detenido el gestor de colas; por ejemplo:

      ```
      C:\Users\Administrator>dspmq -m QMGR
      QMNAME(QMGR) STATUS(Ended immediately)
      ```

   b) En *venus*, ejecute el mandato **strmqm** para iniciar *QMGR* permitiendo esperas:

      ```
      strmqm -x QMGR
      ```

      La respuesta del sistema:

      ```
      IBM MQ queue manager 'QMGR' starting.
      The queue manager is associated with installation 'Installation1'.
      5 log records accessed on queue manager 'QMGR' during the log
      replay phase.
      Log replay for queue manager 'QMGR' complete.
      Transaction manager state recovered for queue manager 'QMGR'.
      IBM MQ queue manager 'QMGR' started using V7.1.0.0.
      ```

Resultados

Para comprobar si el gestor de colas multiinstancia conmuta, realice los pasos siguientes:

1. En *mars*, ejecute el mandato **strmqm** para iniciar *QMGR* permitiendo standbys:

   ```
   strmqm -x QMGR
   ```

   La respuesta del sistema:

   ```
   IBM MQ queue manager 'QMGR' starting.
   The queue manager is associated with installation 'Installation1'.
   A standby instance of queue manager 'QMGR' has been started.
   The active instance is running elsewhere.
   ```

2. En *venus* ejecute el mandato **endmqm**:

   ```
   endmqm -r -s -i QMGR
   ```

   La respuesta del sistema en *venus*:

   ```
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ended, permitting switchover to
   a standby instance.
   ```

Y en *mars*:

```
dspmq
QMNAME(QMGR) STATUS(Running as standby)
C:\Users\wmquser2>dspmq
QMNAME(QMGR) STATUS(Running as standby)
C:\Users\wmquser2>dspmq
QMNAME(QMGR)  STATUS(Running)
```

Qué hacer a continuación

Para verificar un gestor de colas multiinstancia utilizando programas de ejemplo, consulte "Verificación del gestor de colas multiinstancia en Windows" en la página 559.

Windows *Creación de un dominio de Active Directory y DNS en Windows*

Esta tarea crea el dominio *wmq.example.com* en un controlador de dominio Windows 2008 denominado *sun*. Configura el grupo global `Domain mqm` en el dominio, con los derechos correctos y con un usuario.

En una configuración a escala de producción, puede que deba ajustar la configuración a un dominio existente. Por ejemplo, podría definir diferentes grupos de dominio para autorizar diferentes unidades compartidas y para agrupar los ID de usuario que ejecutan gestores de colas.

La configuración del ejemplo consta de tres servidores:

sun
: Un controlador de dominio Windows Server 2008. Es propietario del dominio *wmq.example.com* que contiene *Sun*, *mars* y *venus*. Para ilustrar esto, también se utiliza el servidor de archivos.

mars
: Un Windows Server 2008 utilizado como primer servidor de IBM MQ. Contiene una instancia del gestor de colas de varias instancias denominado *QMGR*.

venus
: Un Windows Server 2008 utilizado como segundo servidor de IBM MQ. Contiene la segunda instancia del gestor de colas de varias instancias denominado *QMGR*.

Sustituya los nombres en cursiva del ejemplo, por los nombres que desee.

Antes de empezar

1. Los pasos de la tarea son coherentes con un Windows Server 2008 que está instalado pero no configurado con ningún rol. Si está configurando un controlador de dominio existente, puede resultar útil intentar los pasos en un nuevo Windows Server 2008. Puede adaptar los pasos a su dominio.

Acerca de esta tarea

En esta tarea, cree un dominio de Active Directory y DNS en un nuevo controlador de dominio. A continuación, configúrelo para que esté listo para instalar IBM MQ en otros servidores y estaciones de trabajo que unen el dominio. Siga la tarea si no está familiarizado con la instalación y la configuración de Active Directory para crear un dominio de Windows. Para poder crear una configuración de gestor de colas multiinstancia, debe crear un dominio de Windows. La tarea no está pensada para ayudarle de la mejor manera posible a configurar un dominio de Windows. Para desplegar gestores de colas multiinstancia en un entorno de producción, debe consultar la documentación de Windows.

Durante la tarea, realice los pasos siguientes:

1. Instale Active Directory.
2. Añada un dominio.
3. Añada el domino a DNS.
4. Cree el grupo global `Domain mqm` y dele los derechos correctos.

5. Añada un usuario y haga que sea miembro del grupo global `Domain mqm`.

Esta tarea es una de un conjunto de tareas relacionadas que ilustran cómo acceder a los datos del gestor de colas y a los archivos de registro. Las tareas muestran cómo crear un gestor de colas con autorización para leer y grabar archivos de datos y registros que están almacenados en un directorio que elija. Acompañan a la tarea, "Dominios de Windows y gestores de colas multiinstancia" en la página 534.

A efectos de la tarea, el nombre de host del controlador de dominio es *sun* y los dos servidores IBM MQ se denominan *mars* y *venus*. El dominio se denomina *wmq.example.com*. Puede sustituir todos los nombres en cursiva en la tarea por nombres de su propia elección.

Procedimiento

1. Inicie la sesión en el controlador de dominio, *sun*, como administrador local o `Workgroup`.

   Si el servidor ya está configurado como controlador de dominio, debe iniciar la sesión como administrador de dominio.
2. Ejecute el asistente de servicios de dominio de Active Directory.
   a) Pulse **Inicio** > **Ejecutar...** Escriba `dcpromo` y pulse **Aceptar**.

   Si los archivos binarios de Active Directory todavía no están instalados, Windows instala los archivos automáticamente.
3. En la primera ventana del asistente, deje el recuadro de selección **Usar la instalación en modo avanzado** sin marcar. Pulse **Siguiente** > **Siguiente** y pulse **Crear un dominio nuevo en un bosque nuevo** > **Siguiente**.
4. Escriba *wmq.example.com* en el campo **FQDN del dominio raíz de bosque** . Pulse **Siguiente**.
5. En la ventana Establecer el nivel funcional del bosque, seleccione **Windows Server 2003**, o posterior, en la lista de **Niveles funcionales del bosque** > **Siguiente**.

   El nivel más antiguo de Windows Server que está soportado por IBM MQ es Windows Server 2003.
6. Opcional: En la ventana Establecer el nivel funcional del dominio, seleccione **Windows Server 2003**, o posterior, en la lista de **Niveles funcionales del dominio** > **Siguiente**.

   Este paso sólo es necesario si establece el nivel funcional del bosque en **Windows Server 2003**.
7. Se abre la ventana Opciones adicionales del controlador de dominio con **Servidor DNS** seleccionado como opción adicional. Pulse **Siguiente** y **Sí** para borrar la ventana de aviso.

   Consejo: Si ya tiene instalado un Servidor DNS, esta opción no aparece. Si desea seguir esta tarea de forma precisa, elimine todos los roles de este controlador de dominio y vuelva a empezar.
8. Deje los directorios `Database`, `Log Files` y `SYSVOL` sin cambios; pulse **Siguiente**.
9. Escriba una contraseña en los campos **Contraseña** y **Confirmar contraseña** de la ventana Contraseña de administrador del modo de restauración de servicios de directorio. Pulse **Siguiente** > **Siguiente**. Seleccione **Reiniciar al completar** en la ventana del asistente final.
10. Cuando el controlador de dominio se reinicie, inicie la sesión como *wmq*`\Administrator`.

    El administrador de servidores se inicia automáticamente.
11. Abra la carpeta *wmq.example.com*`\Users`.
    a) Abra **Gestor de servidores** > **Roles** > **Servicios de dominio de Active Directory** > ***wmq.example.com*** > **Usuarios**.
12. Pulse con el botón derecho del ratón en **Usuarios** > **Nuevo** > **Grupo**.
    a) Escriba un nombre de grupo en el campo **Nombre de grupo**.

       Nota: El nombre de grupo preferido es `Domain mqm`. Escríbalo tal como aparece.

       - Si llama al grupo `Domain mqm` se modifica el comportamiento del Prepare IBM MQ Wizard en una estación de trabajo o servidor de dominio. Hace que Prepare IBM MQ Wizard añada automáticamente el grupo `Domain mqm` al grupo `mqm` local en cada nueva instalación de IBM MQ en el dominio.

- Puede instalar estaciones de trabajo o servidores en un dominio sin grupo global `Domain mqm`. Si lo hace, debe definir un grupo con las mismas propiedades que el grupo `Domain mqm`. Debe hacer que este grupo o los usuarios que son miembros del mismo, sean miembros del grupo `mqm` local siempre que IBM MQ esté instalado en un dominio. Puede colocar usuarios de dominio en grupos múltiples. Cree grupos de dominio múltiples, donde cada grupo corresponde a un conjunto de instalaciones que desea gestionar por separado. Divida los usuarios de dominio, según las instalaciones que gestionan, en diferentes grupos de dominio. Añada cada grupo o grupos de dominio al grupo `mqm` local de distintas instalaciones de IBM MQ. Sólo los usuarios de dominio de los grupos de dominio que son miembros de un grupo `mqm` local específico pueden crear, administrar y ejecutar gestores de colas para dicha instalación.
- El usuario de dominio que designe al instalar IBM MQ en una estación de trabajo o servidor en un dominio debe ser miembro del grupo `Domain mqm` , o de un grupo alternativo que haya definido con las mismas propiedades que el grupo `Domain mqm` .

b) Deje **Global** pulsado como el **Ámbito del grupo** o cámbielo por **Universal**. Deje **Seguridad** pulsada como **Tipo de grupo**. Pulse **Aceptar**.

13. Añada los derechos **Allow Read group membership** y **Allow Read groupMembershipSAM** a los derechos del grupo global `Domain mqm` .
    a) En la barra de acciones del Gestor de servidores, pulse **Ver** > **Características avanzadas**
    b) En el árbol de navegación del Gestor de servidores, pulse **Usuarios**
    c) En la ventana Usuarios, pulse con el botón derecho del ratón en **Dominio mqm** > **Propiedades**
    d) Pulse **Seguridad** > **Avanzada** > **Agregar...**. Escriba `Domain mqm` y pulse **Comprobar nombres** > **Aceptar**.

       El campo **Nombre** se rellena previamente con la serie, `Domain mqm (`*`domain name`*`\Domain mqm)`.
    e) Pulse **Propiedades**. En la lista **Aplicar a**, seleccione **Objetos de usuarios descendientes**.
    f) En la lista **Permisos**, seleccione los recuadros de selección **Leer la pertenencia a grupo** y **Leer groupMembershipSAM Permitir**; pulse **Aceptar** > **Aplicar** > **Aceptar** > **Aceptar**.
14. Añada dos o más usuarios al grupo global `Domain mqm` .

    Un usuario, *`wmquser1`* en el ejemplo, ejecuta el servicio IBM MQ y el otro usuario, *`wmquser2`* , se utiliza de forma interactiva.

    Se precisa un usuario de dominio para crear un gestor de colas que utiliza el grupo de seguridad alternativo en una configuración de dominio. No es suficiente que el ID de usuario sea un administrador, aunque un administrador tiene la autorización para ejecutar el mandato **`crtmqm`**. El usuario de dominio, que podría ser un administrador, debe ser miembro del grupo `mqm` local, así como del grupo de seguridad alternativo.

    En el ejemplo, puede convertir *`wmquser1`* y *`wmquser2`* en miembros del grupo global `Domain mqm` . El Prepare IBM MQ Wizard configura automáticamente `Domain mqm` como miembro del grupo `mqm` local donde se ejecuta el asistente.

    Debe proporcionar un usuario diferente para ejecutar el servicio de IBM MQ para cada instalación de IBM MQ en un solo sistema. Puede reutilizar los mismos usuarios en diferentes sistemas.

    a) En el árbol de navegación del Gestor de servidores, pulse **Usuarios** > **Nuevo** > **Usuario**
    b) En la ventana Objeto nuevo-Usuario, escriba *`wmquser1`* en el campo **Nombre de inicio de sesión de usuario** . Escriba *`WebSphere`* en el campo **Nombre** y *`MQ1`* en el campo **Apellido** . Pulse **Siguiente**.
    c) Escriba una contraseña en los campos **Contraseña** y **Confirmar contraseña** y borre el recuadro de selección **El usuario debe cambiar la contraseña en el siguiente inicio de sesión**. Pulse **Siguiente** > **Finalizar**.
    d) En la ventana Usuarios, pulse con el botón derecho del ratón en ***WebSphere MQ*** > **Añadir a un grupo...**. Escriba `Domain mqm` y pulse **Comprobar nombres** > **Aceptar** > **Aceptar**.
    e) Repita los pasos a a d para añadir *`WebSphere MQ2`* como *`wmquser2`* .
15. Ejecución de IBM MQ como servicio.

Si tiene que ejecutar IBM MQ como servicio y luego otorgar al usuario de dominio (que ha obtenido del administrador de dominio) el acceso para ejecutar como servicio, lleve a cabo el procedimiento siguiente:

a) Pulse **Iniciar > Ejecutar...**.

   Escriba el mandato `secpol.msc` y pulse **Aceptar**.

b) Abra **Configuración de seguridad > Políticas locales > Asignaciones de derechos de usuario**.

   En la lista de políticas, pulse con el botón derecho del ratón **Iniciar sesión como servicio > Propiedades**.

c) Pulse **Añadir usuario o grupo...**

   Escriba el nombre de usuario que ha obtenido del administrador de dominios y pulse **Comprobar nombres**

d) Si se le solicita en una ventana de seguridad de Windows, escriba el nombre de usuario y la contraseña de un usuario o administrador de cuentas con autorización suficiente y pulse **Aceptar > Aplicar > Aceptar**.

   Cierre la ventana Política de seguridad local.

Nota: En Windows Server 2008 y Windows Server 2012 el Control de cuentas de usuario (UAC) está habilitado de forma predeterminada.

La característica UAC restringe las acciones que los usuarios pueden llevar a cabo en determinados recursos del sistema operativo, incluso si son miembros del grupo Administradores. Debe tomar las medidas apropiadas para superar esta restricción.

Qué hacer a continuación

Continúe con la tarea siguiente, "Instalar IBM MQ en un servidor o una estación de trabajo de un dominio de Windows" en la página 542.

Tareas relacionadas

Windows Instalar IBM MQ en un servidor o una estación de trabajo de un dominio de Windows

Windows Creación de un directorio compartido para los archivos de datos y de registro del gestor de colas en Windows

Windows Leer y grabar archivos de datos y de registro compartidos autorizados por un grupo de seguridad global alternativo

Windows *Instalar IBM MQ en un servidor o una estación de trabajo de un dominio de Windows*

En esta tarea, instale y configure IBM MQ en un servidor o estación de trabajo en el dominio *wmq.example.com* Windows.

En una configuración a escala de producción, puede que deba ajustar la configuración a un dominio existente. Por ejemplo, podría definir diferentes grupos de dominio para autorizar diferentes unidades compartidas y para agrupar los ID de usuario que ejecutan gestores de colas.

La configuración del ejemplo consta de tres servidores:

sun
: Un controlador de dominio Windows Server 2008. Es propietario del dominio *wmq.example.com* que contiene *Sun*, *mars* y *venus*. Para ilustrar esto, también se utiliza el servidor de archivos.

mars
: Un Windows Server 2008 utilizado como primer servidor de IBM MQ. Contiene una instancia del gestor de colas de varias instancias denominado *QMGR*.

venus
: Un Windows Server 2008 utilizado como segundo servidor de IBM MQ. Contiene la segunda instancia del gestor de colas de varias instancias denominado *QMGR*.

Sustituya los nombres en cursiva del ejemplo, por los nombres que desee.

Antes de empezar

Importante: De forma predeterminada, los sistemas que empiezan por Windows 10 versión 1607 y Windows Server 2016 son más restrictivos que las versiones anteriores de Windows.

Este cambio restringe los clientes que pueden realizar llamadas remotas al Gestor de cuentas de seguridad (SAM) y podría afectar a IBM MQ con gestores de colas que no se pueden iniciar. El acceso a SAM es fundamental para el funcionamiento de IBM MQ cuando IBM MQ se configura como una cuenta de dominio.

1. Realice los pasos en "Creación de un dominio de Active Directory y DNS en Windows" en la página 539 para crear un controlador de dominio, *sun*, para el dominio *wmq.example.com*. Cambie los nombres en cursiva para ajustarse a su configuración.
2. Consulte Requisitos de hardware y software en sistemas Windows para ver otras versiones de Windows en las que puede ejecutar IBM MQ.

Acerca de esta tarea

En esta tarea se configura un Windows Server 2008, denominado *mars*, como miembro del dominio *wmq.example.com*. Instale IBM MQ y configure la instalación para que se ejecute como miembro del dominio *wmq.example.com*.

Esta tarea es una de un conjunto de tareas relacionadas que ilustran cómo acceder a los datos del gestor de colas y a los archivos de registro. Las tareas muestran cómo crear un gestor de colas con autorización para leer y grabar archivos de datos y registros que están almacenados en un directorio que elija. Acompañan a la tarea, "Dominios de Windows y gestores de colas multiinstancia" en la página 534.

A efectos de la tarea, el nombre de host del controlador de dominio es *sun* y los dos servidores IBM MQ se denominan *mars* y *venus*. El dominio se denomina *wmq.example.com*. Puede sustituir todos los nombres en cursiva en la tarea por nombres de su propia elección.

Procedimiento

1. Añada el controlador de dominio, *sun.wmq.example.com* a *mars* como servidor DNS.
   a) En *mars*, inicie la sesión como *mars*\Administrator y pulse **Inicio**.
   b) Pulse con el botón derecho del ratón en **Red** > **Propiedades** > **Gestionar conexiones de red**.
   c) Pulse con el botón derecho del ratón en el adaptador de red, pulse **Propiedades**.

      El sistema responde con la ventana Propiedades de conexión de área local que lista elementos que la conexión utiliza.

   d) Seleccione **Protocolo Internet Versión 4** o **Protocolo Internet IBM WebSphere MQ 6** en la lista de elementos de la ventana Propiedades de conexión de área local. Pulse **Propiedades** > **Avanzadas ...** y pulse el separador **DNS** .
   e) En las direcciones del servidor DNS, pulse **Añadir...**.
   f) Escriba la dirección IP del controlador de dominio, que también es el servidor DNS y pulse **Añadir**.
   g) Pulse **Añadir estos sufijos DNS** > **Añadir...**.
   h) Escriba *wmq.example.com* y pulse **Agregar**.
   i) Escriba *wmq.example.com* en el campo **Sufijo DNS para esta conexión**.
   j) Seleccione **Registrar la dirección de esta conexión en DNS** y **Utilizar este sufijo de conexión en el registro DNS**. Pulse **Aceptar** > **Aceptar** > **Cerrar**
   k) Abra una ventana de mandatos y escriba el mandato **ipconfig /all** para revisar los valores de TCP/IP.
2. En *mars*, añada el sistema al dominio *wmq.example.com*.
   a) Pulse **Iniciar**

b) Pulse con el botón derecho en **Sistema** > **Propiedades**. En el apartado Configuración de nombre, dominio y grupo de trabajo del equipo, pulse **Cambiar configuración**.

c) En la ventana Propiedades del sistema, pulse **Cambiar...**.

d) Pulse Dominio, escriba *wmq.example.com* y pulse **Aceptar**.

e) Escriba el **Nombre de usuario** y la **Contraseña** del administrador del controlador de dominio, que tiene autorización para permitir que el sistema se una al dominio y pulse **Aceptar**.

f) Pulse **Aceptar** > **Aceptar** > **Cerrar** > **Reiniciar ahora** en respuesta al mensaje "Bienvenido al dominio *wmq.example.com*".

3. Compruebe que el sistema es miembro del dominio *wmq.example.com*

a) En *sun*, inicie la sesión en el controlador de dominio como *wmq*\Administrator.

b) Abra **Administrador del servidor** > **Active Directory Servicios de dominio** > ***wmq.example.com*** > **Equipos** y compruebe que *mars* aparezca listado correctamente en la ventana Equipos.

4. Instale IBM MQ for Windows en *mars*.

Si desea información adicional sobre cómo ejecutar el asistente de instalación de IBM MQ for Windows; consulte Instalación del servidor de IBM MQ en Windows .

a) En *mars*, inicie la sesión como administrador local, *mars*\Administrator.

b) Ejecute el mandato **Setup** en el soporte de instalación de IBM MQ for Windows.

Se iniciará la aplicación Launchpad de IBM MQ.

c) Pulse **Requisitos de software** para comprobar si el software de requisitos previos está instalado.

d) Pulse **configuración de red** > **Sí** para configurar un ID de usuario de dominio.

La tarea, “Creación de un dominio de Active Directory y DNS en Windows” en la página 539, configura un ID de usuario de dominio para este conjunto de tareas.

e) Pulse **Instalación de IBM MQ**, seleccione un idioma de instalación y pulse Iniciar instalador de IBM MQ.

f) Confirme el acuerdo de licencia y pulse **Siguiente** > **Siguiente** > **Instalar** para aceptar la configuración predeterminada. Espere hasta que se complete la instalación y pulse **Finalizar**.

Si desea cambiar el nombre de la instalación, instale otros componentes, configure un directorio diferente para datos y registros del gestor de colas o realice la instalación en un directorio diferente. De esta manera, pulse **Personalizada** en lugar de **Típica**.

Se instala IBM MQ y el instalador inicia el Prepare IBM MQ Wizard.

Importante: No ejecute todavía el asistente.

5. Configure el usuario que va a ejecutar el servicio de IBM MQ con el derecho **Ejecutar como servicio**.

Elija si desea configurar el grupo mqm local, el grupo Domain mqm o el usuario que va a ejecutar el servicio IBM MQ con el derecho. En el ejemplo, da al usuario el derecho.

a) Pulse **Iniciar** > **Ejecutar ...**, escriba el mandato **secpol.msc** y pulse **Aceptar**.

b) Abrir **Configuración de seguridad** > **Políticas locales** > **Asignaciones de derechos de usuario**. En la lista de políticas, pulse con el botón derecho del ratón en **Iniciar sesión como servicio** > **Propiedades** .

c) Pulse **Añadir usuario o grupo...** y escriba *wmquser1* y pulse **Comprobar nombres**

d) Escriba el nombre de usuario y la contraseña de un administrador de dominio, *wmq*\Administrator, y pulse **Aceptar** > **Aplicar** > **Aceptar**. Cierre la ventana Política de seguridad local.

6. Ejecute Prepare IBM MQ Wizard.

Si desea más información, consulte Configuración de IBM MQ con el Prepare IBM MQ Wizard.

a) El instalador de IBM MQ ejecuta el Prepare IBM MQ Wizard automáticamente.

Para iniciar el asistente manualmente, busque el acceso directo al Prepare IBM MQ Wizard en la carpeta **Inicio** > **Todos los programas** > **IBM MQ**. Seleccione el acceso directo que corresponda a la instalación de IBM MQ en una configuración de varias instalaciones.

b) Pulse **Siguiente** y deje **Sí** pulsado en respuesta a la pregunta "Identificar si hay un controlador de dominio de Windows 2000 o posterior en la red".

c) Pulse **Sí** > **Siguiente** en la ventana Configuración de IBM MQ for Windows para usuarios de Windows.

d) En la segunda ventana Configuración de IBM MQ for Windows para usuarios de dominio de Windows , escriba *wmq* en el campo **Dominio** . Escriba *wmquser1* en el campo **Nombre de usuario** y la contraseña, si la ha establecido, en el campo **Contraseña** . Pulse **Siguiente**.

El asistente configura e inicia IBM MQ con *wmquser1*.

e) En la página final del asistente, seleccione o borre los recuadros de selección tal como sea necesario y pulse **Finalizar**.

Qué hacer a continuación

1. Realice la tarea, “Lectura y grabación de datos y archivos de registro autorizados por el grupo mqm local” en la página 566, para verificar que la instalación y configuración están funcionando correctamente.
2. Realice la tarea, “Creación de un directorio compartido para los archivos de datos y de registro del gestor de colas en Windows” en la página 545, para configurar una compartición de archivo para almacenar los archivos de datos y de registro de un gestor de colas multiinstancia.

Tareas relacionadas

Windows Creación de un dominio de Active Directory y DNS en Windows

Windows Creación de un directorio compartido para los archivos de datos y de registro del gestor de colas en Windows

Windows Leer y grabar archivos de datos y de registro compartidos autorizados por un grupo de seguridad global alternativo

Referencia relacionada

Derechos de usuario necesarios para un servicio IBM MQ Windows

Windows *Creación de un directorio compartido para los archivos de datos y de registro del gestor de colas en Windows*

Esta tarea es una de un conjunto de tareas relacionadas que ilustran cómo acceder a los datos del gestor de colas y a los archivos de registro. Las tareas muestran cómo crear un gestor de colas con autorización para leer y grabar archivos de datos y registros que están almacenados en un directorio que elija.

En una configuración a escala de producción, puede que deba ajustar la configuración a un dominio existente. Por ejemplo, podría definir diferentes grupos de dominio para autorizar diferentes unidades compartidas y para agrupar los ID de usuario que ejecutan gestores de colas.

La configuración del ejemplo consta de tres servidores:

sun
Un controlador de dominio Windows Server 2008. Es propietario del dominio *wmq.example.com* que contiene *Sun*, *mars* y *venus*. Para ilustrar esto, también se utiliza el servidor de archivos.

mars
Un Windows Server 2008 utilizado como primer servidor de IBM MQ. Contiene una instancia del gestor de colas de varias instancias denominado *QMGR*.

venus
Un Windows Server 2008 utilizado como segundo servidor de IBM MQ. Contiene la segunda instancia del gestor de colas de varias instancias denominado *QMGR*.

Sustituya los nombres en cursiva del ejemplo, por los nombres que desee.

Antes de empezar

1. Para realizar esta tarea exactamente tal como está documentada, realice los pasos de la tarea, "Creación de un dominio de Active Directory y DNS en Windows" en la página 539, para crear el dominio *sun.wmq.example.com* en el controlador de dominio *sun*. Cambie los nombres en cursiva para ajustarse a su configuración.

Acerca de esta tarea

Esta tarea es una de un conjunto de tareas relacionadas que ilustran cómo acceder a los datos del gestor de colas y a los archivos de registro. Las tareas muestran cómo crear un gestor de colas con autorización para leer y grabar archivos de datos y registros que están almacenados en un directorio que elija. Acompañan a la tarea, "Dominios de Windows y gestores de colas multiinstancia" en la página 534.

En la tarea, cree una unidad compartida que contenga un directorio de datos y registros, así como un grupo local para autorizar el acceso a la unidad compartida. Pase el nombre del grupo global que autoriza el compartimiento al mandato **crtmqm** en su parámetro `-a`. El grupo global le ofrece la flexibilidad de separar los usuarios de esta unidad compartida de los usuarios de otras unidades compartidas. Si no necesita esta flexibilidad, autorice la compartición con el grupo `Domain mqm` en lugar de crear un nuevo grupo global.

El grupo global que se utiliza para compartir en esta tarea se denomina *wmqha* y la compartición se denomina *wmq*. Están definidos en el controlador de dominio *sun* en el dominio de Windows *wmq.example.com*. La compartición tiene permisos de control completos para el grupo global *wmqha*. Sustituya los nombres en cursiva en la tarea por nombres que elija.

Para los fines de esta tarea, el controlador de dominio es el mismo servidor que el servidor de archivos. En aplicaciones prácticas, divida los servicios de directorios y archivos entre diferentes servidores para rendimiento y disponibilidad.

Debe configurar el ID de usuario bajo el que se ejecuta el gestor de colas para que sea miembro de dos grupos. Hay que ser miembro del grupo local `mqm` de un servidor de IBM MQ y del grupo global *wmqha*.

En este conjunto de tareas, cuando el gestor de colas se ejecuta como un servicio, se ejecuta bajo el ID de usuario *wmquser1*, por lo que *wmquser1* debe ser miembro de *wmqha*. Cuando el gestor de colas se ejecuta de forma interactiva, se ejecuta bajo el ID de usuario *wmquser2*, por lo que *wmquser2* debe ser miembro de *wmqha*. Tanto *wmquser1* como *wmquser2* son miembros del grupo global `Domain mqm`. `Domain mqm` es un miembro del grupo `mqm` local en los servidores *mars* y *venus* IBM MQ . Por lo tanto, *wmquser1* y *wmquser2* son miembros del grupo local `mqm` en ambos servidores de IBM MQ.

Procedimiento

1. Inicie la sesión en el controlador de dominio, *sun.wmq.example.com* como administrador del dominio.
2. Cree el grupo global *wmqha*.
   a) Abra **Gestor de servidores** > **Roles** > **Servicios de dominio de Active Directory** > ***wmq.example.com*** > **Usuarios**.
   b) Abra la carpeta *wmq.example.com*`\Users`.
   c) Pulse con el botón derecho del ratón en **Usuarios** > **Nuevo** > **Grupo**.
   d) Escriba *wmqha* en el campo **Nombre de grupo** .
   e) Deje **Global** pulsado como **Ámbito del grupo** y **Seguridad** como **Tipo de grupo**. Pulse **Aceptar**.
3. Añada los usuarios de dominio *wmquser1* y *wmquser2* al grupo global, *wmqha*.
   a) En el árbol de navegación Administrador de servidores, pulse **Usuarios** y pulse con el botón derecho del ratón en ***wmqha*** > **Propiedades** en la lista de usuarios.
   b) Pulse la pestaña Miembros en la ventana Propiedades de *wmqha*.
   c) Pulse **Añadir ...** ; Escriba *wmquser1* ; *wmquser2* y pulse **Comprobar nombres** > **Aceptar** > **Aplicar** > **Aceptar**.

4. Cree el árbol de directorio para contener archivos de datos y registros del gestor de colas.
   a) Abra un indicador de mandatos.
   b) Escriba el mandato:

```
md c:\wmq\data, c:\wmq\logs
```

5. Autorice al grupo global *wmqha* para que tenga permiso de control total sobre los directorios y la compartición *c:\wmq*.
   a) En el Explorador de Windows, pulse con el botón derecho del ratón en ***c:\wmq*** > **Propiedades**.
   b) Pulse la pestaña **Seguridad** y pulse **Avanzada** > **Editar...**.
   c) Borre el recuadro de selección para **Incluir permisos heredados del propietario de este objeto**. Pulse **Copiar** en la ventana Seguridad de Windows.
   d) Seleccione las líneas para usuarios en la lista **Entradas de permiso** y pulse **Eliminar**. Deje las líneas para SYSTEM, Administradores y CREATOR OWNER en la lista de **Entradas de permiso**.
   e) Pulse **Añadir ...**, y escriba el nombre del grupo global *wmqha*. Pulse **Comprobar nombres** > **Aceptar**.
   f) En la ventana Entrada de permiso para wmq, seleccione **Control completo** en la lista de **Permisos**.
   g) Pulse **Aceptar** > **Aplicar** > **Aceptar** > **Aceptar** > **Aceptar**
   h) En Windows Explorer, pulse con el botón derecho del ratón en ***c:\wmq*** > **Compartir ...**.
   i) Pulse **Uso compartido avanzado ...** y marque el recuadro de selección **Compartir esta carpeta** . Deje el nombre de compartición como *wmq*.
   j) Pulse **Permisos** > **Añadir ...**, y escriba el nombre del grupo global *wmqha*. Pulse **Comprobar nombres** > **Aceptar**.
   k) Seleccione *wmqha* en la lista de **Nombres de grupo o de usuario**. Seleccione la casilla **Control completo** en la lista de **Permisos para *wmqha*** ; pulse **Aplicar**.
   l) Seleccione *Administrators* en la lista de **Nombres de grupo o de usuario**. Seleccione la casilla **Control completo** en la lista de **Permisos para *Administradores*** ; pulse **Aplicar** > **Aceptar** > **Aceptar** > **Cerrar**.

Qué hacer a continuación

Compruebe que pueda leer y escribir archivos en los directorios compartidos desde cada uno de los servidores de IBM MQ. Compruebe el ID de usuario de servicio de IBM MQ, *wmquser1* y el ID de usuario interactivo, *wmquser2*.

1. Si está utilizando el escritorio remoto, debe añadir *wmq**wmquser1* y *wmquser2* al grupo local Remote Desktop Users en *mars*.
   a. Inicie la sesión en *mars* como *wmq*\Administrator
   b. Ejecute el mandato **lusrmgr.msc** para abrir la ventana Usuarios locales y grupos.
   c. Pulse **Grupos**. Pulse con el botón derecho del ratón en **Usuarios de escritorio remoto** > **Propiedades** > **Agregar...**. Escriba *wmquser1 ; wmquser2* y pulse **Comprobar nombres**.
   d. Escriba el nombre de usuario y la contraseña del administrador del dominio, *wmq*\Administrator, y pulse **Aceptar** > **Aplicar** > **Aceptar**.
   e. Cierre la ventana Usuarios locales y grupos.
2. Inicie la sesión en *mars* como *wmq**wmquser1*.
   a. Abra una ventana de Windows Explorer y escriba *sun**wmq*.

      El sistema responde abriendo la compartición de *wmq* en *sun.wmq.example.com* y lista los directorios de datos y registros.
   b. Compruebe los permisos de *wmquser1* creando un archivo en el subdirectorio de datos, añadiendo cierto contenido, leyéndolo y, a continuación, suprimiéndolo.

3. Inicie la sesión en *mars* como *wmq\wmquser2* y repita las comprobaciones.
4. Continúe con la siguiente tarea para crear un gestor de colas para utilizar los directorios de datos y registros; consulte "Leer y grabar archivos de datos y de registro compartidos autorizados por un grupo de seguridad global alternativo" en la página 548.

Tareas relacionadas

Windows Creación de un dominio de Active Directory y DNS en Windows

Windows Instalar IBM MQ en un servidor o una estación de trabajo de un dominio de Windows

Windows Leer y grabar archivos de datos y de registro compartidos autorizados por un grupo de seguridad global alternativo

Windows *Leer y grabar archivos de datos y de registro compartidos autorizados por un grupo de seguridad global alternativo*

Esta tarea muestra cómo utilizar el distintivo -a en el mandato **crtmqm**. El distintivo -a otorga acceso del gestor de colas a sus archivos de registro y de datos en un compartimiento de archivos remotos utilizando el grupo de seguridad alternativo.

En una configuración a escala de producción, puede que deba ajustar la configuración a un dominio existente. Por ejemplo, podría definir diferentes grupos de dominio para autorizar diferentes unidades compartidas y para agrupar los ID de usuario que ejecutan gestores de colas.

La configuración del ejemplo consta de tres servidores:

sun
Un controlador de dominio Windows Server 2008. Es propietario del dominio *wmq.example.com* que contiene *Sun*, *mars* y *venus*. Para ilustrar esto, también se utiliza el servidor de archivos.

mars
Un Windows Server 2008 utilizado como primer servidor de IBM MQ. Contiene una instancia del gestor de colas de varias instancias denominado *QMGR*.

venus
Un Windows Server 2008 utilizado como segundo servidor de IBM MQ. Contiene la segunda instancia del gestor de colas de varias instancias denominado *QMGR*.

Sustituya los nombres en cursiva del ejemplo, por los nombres que desee.

Antes de empezar

Efectúe los pasos de las tareas siguientes. Las tareas crean el controlador de dominio y el dominio, instalan IBM MQ for Windows en un servidor y crean la compartición de archivos para los datos y los archivos de registro. Si está configurando un controlador de dominio existente, puede resultar útil intentar los pasos en un nuevo Windows Server 2008. Puede adaptar los pasos a su dominio.

1. "Creación de un dominio de Active Directory y DNS en Windows" en la página 539.
2. "Instalar IBM MQ en un servidor o una estación de trabajo de un dominio de Windows" en la página 542.
3. "Creación de un directorio compartido para los archivos de datos y de registro del gestor de colas en Windows" en la página 545.

Acerca de esta tarea

Esta tarea es una de un conjunto de tareas relacionadas que ilustran cómo acceder a los datos del gestor de colas y a los archivos de registro. Las tareas muestran cómo crear un gestor de colas con autorización para leer y grabar archivos de datos y registros que están almacenados en un directorio que elija. Acompañan a la tarea, "Dominios de Windows y gestores de colas multiinstancia" en la página 534.

En esta tarea, se crea un gestor de colas que almacena sus datos y registros en un directorio remoto de un servidor de archivos. Para la finalidad de este ejemplo, el servidor de archivos es el mismo servidor

que el controlador de dominio. El directorio que contiene las carpetas de datos y de registro se comparte con el permiso de control completo proporcionado al grupo global wmqha.

Procedimiento

1. Inicie la sesión en el servidor de dominio, *mars*, como administrador local, *mars*\Administrator.
2. Abra una ventana de mandatos.
3. Reinicie el servicio IBM MQ.

   Debe reiniciar el servicio para que el ID de usuario bajo el que se ejecuta adquiera las credenciales de seguridad adicionales que ha configurado para ello.

   Escriba los mandatos:

   ```
   endmqsvc
   strmqsvc
   ```

   Las respuestas del sistema:

   ```
   5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
   The MQ service for installation 'Installation1' ended successfully.
   ```

   Y:

   ```
   5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
   The MQ service for installation 'Installation1' started successfully.
   ```
4. Cree el gestor de colas.

   ```
   crtmqm -a wmq\wmqha -sax -u SYSTEM.DEAD.LETTER.QUEUE -md \\sun\wmq\data -ld \\sun\wmq\logs
   QMGR
   ```

   Debe especificar el dominio, *wmq*, del grupo de seguridad alternativo *wmqha* especificando el nombre de dominio completo del grupo global *"wmq\wmqha"*.

   Debe especificar el nombre UNC (Universal Naming Convention) de la compartición *\\sun\wmq* y no utilizar una referencia de unidad correlacionada.

   La respuesta del sistema:

   ```
   IBM MQ queue manager created.
   Directory '\\sun\wmq\data\QMGR' created.
   The queue manager is associated with installation '1'
   Creating or replacing default objects for queue manager 'QMGR'
   Default objects statistics : 74 created. 0 replaced.
   Completing setup.
   Setup completed.
   ```

Qué hacer a continuación

Pruebe el gestor de colas transfiriendo y obteniendo un mensaje de una cola.

1. Inicie el gestor de colas.

   ```
   strmqm QMGR
   ```

   La respuesta del sistema:

```
IBM MQ queue manager 'QMGR' starting.
The queue manager is associated with installation '1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
IBM MQ queue manager 'QMGR' started using V7.1.0.0.
```

2. Cree una cola de prueba.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

La respuesta del sistema:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
Starting MQSC for queue manager QMGR.


1 : define qlocal(QTEST)
AMQ8006: IBM MQ queue created.
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

3. Transfiera un mensaje de prueba utilizando el programa de ejemplo **amqsput**.

```
echo 'A test message' | amqsput QTEST QMGR
```

La respuesta del sistema:

```
Sample AMQSPUT0 start
target queue is QTEST
Sample AMQSPUT0 end
```

4. Obtenga el mensaje de prueba utilizando el programa de ejemplo **amqsget**.

```
amqsget QTEST QMGR
```

La respuesta del sistema:

```
Sample AMQSGET0 start
message A test message
Wait 15 seconds ...
no more messages
Sample AMQSGET0 end
```

5. Detenga el gestor de colas.

```
endmqm -i QMGR
```

La respuesta del sistema:

```
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ended.
```

6. Suprima el gestor de colas.

```
dltmqm QMGR
```

   La respuesta del sistema:

```
IBM MQ queue manager 'QMGR' deleted.
```

7. Suprima los directorios que ha creado.

   Consejo: Añada la opción /Q a los mandatos para impedir que el mandato solicite la supresión de cada archivo o directorio.

```
del /F /S C:\wmq\*.*
rmdir /S C:\wmq
```

Tareas relacionadas

Windows Creación de un dominio de Active Directory y DNS en Windows

Windows Instalar IBM MQ en un servidor o una estación de trabajo de un dominio de Windows

Windows Creación de un directorio compartido para los archivos de datos y de registro del gestor de colas en Windows

Windows *Creación de un gestor de colas multiinstancia en controladores de dominio de Windows*

Un ejemplo muestra cómo configurar un gestor de colas multiinstancia en controladores de dominio de Windows. La configuración muestra los conceptos implicados en vez de realizarse a una escala de producción. El ejemplo se basa en Windows Server 2008. Los pasos pueden ser diferentes en otras versiones de Windows Server.

La configuración utiliza el concepto de minidominio o "domainlet"; consulte Nodos de clúster de Windows 2000, Windows Server 2003 y Windows Server 2008 como controladores de dominio. Para añadir gestores de colas multiinstancia a un dominio existente, consulte “Creación de gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio en Windows” en la página 536.

La configuración del ejemplo consta de tres servidores:

sun
: Un servidor Windows Server 2008 utilizado como primer controlador de dominio. Define el dominio *wmq.example.com* que contiene *sun*, *earth* y *mars* Contiene una instancia del gestor de colas de varias instancias denominado *QMGR*.

earth
: Un Windows Server 2008 utilizado como segundo servidor de controlador de dominio de IBM MQ. Contiene la segunda instancia del gestor de colas de varias instancias denominado *QMGR*.

mars
: Un Windows Server 2008 utilizado como servidor de archivos.

Sustituya los nombres en cursiva del ejemplo, por los nombres que desee.

Antes de empezar

1. En Windows, no es necesario verificar el sistema de archivos en el que tiene pensado guardar los archivos de datos y registros del gestor de colas. El procedimiento de comprobación, Verificación del comportamiento del sistema de archivos compartidos, es aplicable a AIX and Linux. En Windows, las comprobaciones siempre son satisfactorias.
2. Siga los pasos que se indican en el apartado “Creación de un dominio de Active Directory y DNS en Windows” en la página 539 para crear el primero controlador de dominio.

3. Siga los pasos que se indican en el apartado "Adición de un segundo controlador de dominio de Windows a un dominio de ejemplo" en la página 555 para añadir un segundo controlador de dominio, instalar IBM MQ for Windows en ambos controladores de domino y verificar las instalaciones.
4. Siga los pasos que se indican en el apartado "Instalación de IBM MQ en controladores de dominio de Windows en un dominio de ejemplo" en la página 556 para instalar IBM MQ en los dos controladores de dominio.

Acerca de esta tarea

En un servidor de archivos del mismo dominio, cree una unidad compartida para los directorios de datos y de registros del gestor de colas. A continuación, cree la primera instancia de un gestor de colas multiinstancia que utiliza la compartición de archivos en uno de los controladores de dominio. Cree la otra instancia en el otro controlador de dominio y, finalmente, verifique la configuración. Puede crear la compartición de archivos un controlador de dominio.

En el ejemplo, *sun* es el primer controlador de dominio, *earth* el segundo y *mars* es el servidor de archivos.

Procedimiento

1. Cree los directorios que contendrán los archivos de registros y datos del gestor de colas.
   a) En *mars*, escriba el mandato:

```
md c:\wmq\data , c:\wmq\logs
```

2. Comparta los directorios que contendrán los archivos de registros y datos del gestor de colas.

   Debe permitir el acceso de control completo al grupo local de dominio `mqm` y al ID de usuario que utilice para crear el gestor de colas. En el ejemplo, los ID de usuario que son miembros de `Domain Administrators` tienen autorización para crear gestores de colas.

   La compartición de archivos debe ser en un servidor que esté en el mismo dominio que los controladores de dominio. En el ejemplo, el servidor *mars* está en el mismo dominio que los controladores de dominio.

   a) En el Explorador de Windows, pulse con el botón derecho del ratón en ***c:\wmq*** > **Propiedades**.
   b) Pulse la pestaña **Seguridad** y pulse **Avanzada** > **Editar...**.
   c) Borre el recuadro de selección para **Incluir permisos heredados del propietario de este objeto**. Pulse **Copiar** en la ventana Seguridad de Windows.
   d) Seleccione las líneas para usuarios en la lista **Entradas de permiso** y pulse **Eliminar**. Deje las líneas para SYSTEM, Administradores y CREATOR OWNER en la lista de **Entradas de permiso**.
   e) Pulse **Añadir ...**, y escriba el nombre del grupo local de dominio *mqm*. Pulse **Comprobar nombres**
   f) En respuesta a una ventana de Windows Security, escriba el nombre y la contraseña de `Domain Administrator` y pulse **Aceptar** > **Aceptar**.
   g) En la ventana Entrada de permiso para `wmq`, seleccione **Control completo** en la lista de **Permisos**.
   h) Pulse **Aceptar** > **Aplicar** > **Aceptar** > **Aceptar** > **Aceptar**
   i) Repita los pasos e a h para añadir `Domain Administrators`.
   j) En Windows Explorer, pulse con el botón derecho del ratón en ***c:\wmq*** > **Compartir ...**.
   k) Pulse **Uso compartido avanzado ...** y marque el recuadro de selección **Compartir esta carpeta** . Deje el nombre de compartición como *wmq*.
   l) Pulse **Permisos** > **Añadir ...**, y escriba el nombre del grupo local de dominio *mqm* ; `Domain Administrators`. Pulse **Comprobar nombres**.
   m) En respuesta a una ventana de Windows Security, escriba el nombre y la contraseña de `Domain Administrator` y pulse **Aceptar** > **Aceptar**.
3. Cree el gestor de colas *QMGR* en el primer controlador de dominio, *sun*.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md \\mars\wmq\data -ld \\mars\wmq\logs QMGR
```

La respuesta del sistema:

```
IBM MQ queue manager created.
Directory '\\mars\wmq\data\QMGR' created.
The queue manager is associated with installation 'Installation1'.
Creating or replacing default objects for queue manager 'QMGR'.
Default objects statistics : 74 created. 0 replaced. 0 failed.
Completing setup.
Setup completed.
```

4. Inicie el gestor de colas en *sun*, lo que permite una instancia en espera.

```
strmqm -x QMGR
```

La respuesta del sistema:

```
IBM MQ queue manager 'QMGR' starting.
The queue manager is associated with installation 'Installation1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
IBM MQ queue manager 'QMGR' started using V7.1.0.0.
```

5. Cree una segunda instancia de *QMGR* en *earth*.
   a) Compruebe que los valores de los parámetros `Prefix` e `InstallationName` son correctos para *earth*.

      En *sun*, ejecute el mandato **dspmqinf**:

      ```
      dspmqinf QMGR
      ```

      La respuesta del sistema:

      ```
      QueueManager:
      Name=QMGR
      Directory=QMGR
      Prefix= C:\ProgramData \IBM \MQ
      DataPath=\\mars\wmq\data\QMGR
      InstallationName=Installation1
      ```

   b) Copie el formato legible por máquina de la stanza **QueueManager** en el portapapeles.

      En *sun* ejecute de nuevo el mandato **dspmqinf**, con el parámetro `-o`.

      ```
      dspmqinf -o command QMGR
      ```

      La respuesta del sistema:

      ```
      addmqinf -s QueueManager -v Name=QMGR
      -v Directory=QMGR -v Prefix=" C:\ProgramData \IBM \MQ"
      -v DataPath=\\mars\wmq\data\QMGR
      ```

c) En *earth* ejecute el mandato **addmqinf** del portapapeles para crear una instancia del gestor de colas en *earth*.

Ajuste el mandato si es necesario, para acomodar las diferencias en los parámetros `Prefix` o `InstallationName`.

```
addmqinf -s QueueManager -v Name= QMGR
-v Directory= QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"
-v DataPath=\\mars\wmq\data\QMGR
```

```
IBM MQ configuration information added.
```

6. Inicie la instancia en espera del gestor de colas en *earth*.

```
strmqm -x QMGR
```

La respuesta del sistema:

```
IBM MQ queue manager 'QMGR' starting.
The queue manager is associated with installation 'Installation1'.
A standby instance of queue manager 'QMGR' has been started. The active
instance is running elsewhere.
```

Resultados

Verifique que el gestor de colas pasa de *sun* a *earth*:

1. En *sun*, ejecute el mandato:

```
endmqm -i -r -s QMGR
```

La respuesta del sistema en *sun*:

```
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ended, permitting switchover to
a standby instance.
```

2. En *earth*, escriba repetidamente el mandato:

```
dspmq
```

Las respuestas del sistema:

```
QMNAME(QMGR) STATUS(Running as standby)
QMNAME(QMGR) STATUS(Running as standby)
QMNAME(QMGR) STATUS(Running)
```

Qué hacer a continuación

Para verificar un gestor de colas multiinstancia utilizando programas de ejemplo, consulte “Verificación del gestor de colas multiinstancia en Windows” en la página 559.

Tareas relacionadas

“Adición de un segundo controlador de dominio de Windows a un dominio de ejemplo” en la página 555
“Instalación de IBM MQ en controladores de dominio de Windows en un dominio de ejemplo” en la página 556

Información relacionada

Nodos de clúster de Windows 2000, Windows Server 2003 y Windows Server 2008 como controladores de dominio

Windows *Adición de un segundo controlador de dominio de Windows a un dominio de ejemplo*

Añada un segundo controlador de dominio al dominio *wmq.example.com* para construir un dominio Windows en el que ejecutará gestores de colas de varias instancias en controladores de dominio y servidores de archivos

La configuración del ejemplo consta de tres servidores:

sun
: Un servidor Windows Server 2008 utilizado como primer controlador de dominio. Define el dominio *wmq.example.com* que contiene *sun*, *earth* y *mars* Contiene una instancia del gestor de colas de varias instancias denominado *QMGR*.

earth
: Un Windows Server 2008 utilizado como segundo servidor de controlador de dominio de IBM MQ. Contiene la segunda instancia del gestor de colas de varias instancias denominado *QMGR*.

mars
: Un Windows Server 2008 utilizado como servidor de archivos.

Sustituya los nombres en cursiva del ejemplo, por los nombres que desee.

Antes de empezar

1. Realice los pasos en “Creación de un dominio de Active Directory y DNS en Windows” en la página 539 para crear un controlador de dominio, *sun*, para el dominio *wmq.example.com*. Cambie los nombres en cursiva para ajustarse a su configuración.
2. Instale Windows Server 2008 en un servidor en el grupo de trabajo predeterminado, WORKGROUP. Para el ejemplo, el servidor se denomina *earth*.

Acerca de esta tarea

En esta tarea se configura un Windows Server 2008, denominado *earth*, como segundo controlador de dominio en el dominio *wmq.example.com*.

Esta tarea es una de un conjunto de tareas relacionadas que ilustran cómo acceder a los datos del gestor de colas y a los archivos de registro. Las tareas muestran cómo crear un gestor de colas con autorización para leer y grabar archivos de datos y registros que están almacenados en un directorio que elija. Acompañan a la tarea, “Dominios de Windows y gestores de colas multiinstancia” en la página 534.

Procedimiento

1. Añada el controlador de dominio, *sun.wmq.example.com* a *earth* como servidor DNS.
   a) En *earth*, inicie la sesión como *earth*\Administrator y pulse **Inicio**.
   b) Pulse con el botón derecho del ratón en **Red** > **Propiedades** > **Gestionar conexiones de red**.
   c) Pulse con el botón derecho del ratón en el adaptador de red, pulse **Propiedades**.

      El sistema responde con la ventana Propiedades de conexión de área local que lista elementos que la conexión utiliza.

   d) Seleccione **Protocolo Internet Versión 4** o **Protocolo Internet IBM WebSphere MQ 6** en la lista de elementos de la ventana Propiedades de conexión de área local. Pulse **Propiedades** > **Avanzadas ...** y pulse el separador **DNS** .

e) En las direcciones del servidor DNS, pulse **Añadir...**.

f) Escriba la dirección IP del controlador de dominio, que también es el servidor DNS y pulse **Añadir**.

g) Pulse **Añadir estos sufijos DNS** > **Añadir...**.

h) Escriba *wmq.example.com* y pulse **Agregar**.

i) Escriba *wmq.example.com* en el campo **Sufijo DNS para esta conexión**.

j) Seleccione **Registrar la dirección de esta conexión en DNS** y **Utilizar este sufijo de conexión en el registro DNS**. Pulse **Aceptar** > **Aceptar** > **Cerrar**

k) Abra una ventana de mandatos y escriba el mandato **ipconfig /all** para revisar los valores de TCP/IP.

2. Inicie la sesión en el controlador de dominio, *sun*, como administrador local o `Workgroup`.

   Si el servidor ya está configurado como controlador de dominio, debe iniciar la sesión como administrador de dominio.

3. Ejecute el asistente de servicios de dominio de Active Directory.

   a) Pulse **Inicio** > **Ejecutar...** Escriba `dcpromo` y pulse **Aceptar**.

   Si los archivos binarios de Active Directory todavía no están instalados, Windows instala los archivos automáticamente.

4. Configure *earth* como segundo controlador de dominio en el dominio de *wmq.example.com*.

   a) En la primera ventana del asistente, deje el recuadro de selección **Usar la instalación en modo avanzado** sin marcar. Pulse **Siguiente** > **Siguiente** y pulse **Agregar un controlador de dominio a un dominio existente** > **Siguiente**.

   b) Escriba *wmq* en **Escriba el nombre de cualquier dominio de este bosque ...** . El botón de selección **Credenciales alternativos** está pulsado, pulse **Establecer...**. Escriba el nombre y la contraseña del administrador de dominio y pulse **Aceptar** > **Siguiente** > **Siguiente** > **Siguiente**.

   c) En la ventana Opciones adicionales del controlador de dominio, acepte las opciones **Servidor DNS** y **Catálogo global**, que están seleccionadas; pulse **Siguiente** > **Siguiente**.

   d) En la Contraseña de administrador del modo de restauración de servicios de directorio, escriba una **Contraseña** y **Confirmar contraseña** y pulse **Siguiente** > **Siguiente**.

   e) Cuando se le soliciten **Credenciales de red**, escriba la contraseña del administrador de dominio. Seleccione **Reiniciar al completar** en la ventana del asistente final.

   f) Poco después, es posible que se abra una ventana con un error**DCPromo** relativo a la delegación de DNS; pulse **Aceptar**. El servidor se reinicia.

Resultados

Cuando *earth* se haya reiniciado, inicie la sesión como Administrador de dominio. Compruebe que el dominio `wmq.example.com` se ha replicado en *earth*.

Qué hacer a continuación

Continúe con la instalación de IBM MQ; consulte “Instalación de IBM MQ en controladores de dominio de Windows en un dominio de ejemplo” en la página 556.

Tareas relacionadas

Windows Instalación de IBM MQ en controladores de dominio de Windows en un dominio de ejemplo

“Creación de un dominio de Active Directory y DNS en Windows” en la página 539

Windows *Instalación de IBM MQ en controladores de dominio de Windows en un dominio de ejemplo*

Instale y configure instalaciones de IBM MQ en ambos controladores de dominio en el dominio *wmq.example.com*.

La configuración del ejemplo consta de tres servidores:

sun
Un servidor Windows Server 2008 utilizado como primer controlador de dominio. Define el dominio *wmq.example.com* que contiene *sun*, *earth* y *mars* Contiene una instancia del gestor de colas de varias instancias denominado *QMGR*.

earth
Un Windows Server 2008 utilizado como segundo servidor de controlador de dominio de IBM MQ. Contiene la segunda instancia del gestor de colas de varias instancias denominado *QMGR*.

mars
Un Windows Server 2008 utilizado como servidor de archivos.

Sustituya los nombres en cursiva del ejemplo, por los nombres que desee.

Antes de empezar

1. Realice los pasos en "Creación de un dominio de Active Directory y DNS en Windows" en la página 539 para crear un controlador de dominio, *sun*, para el dominio *wmq.example.com*. Cambie los nombres en cursiva para ajustarse a su configuración.
2. Realice los pasos en "Adición de un segundo controlador de dominio de Windows a un dominio de ejemplo" en la página 555 para crear un segundo controlador de dominio, *earth*, para el dominio *wmq.example.com*. Cambie los nombres en cursiva para ajustarse a su configuración.
3. Consulte Requisitos de hardware y software en sistemas Windows para ver otras versiones de Windows en las que puede ejecutar IBM MQ.

Acerca de esta tarea

Instale y configure instalaciones de IBM MQ en ambos controladores de dominio en el dominio *wmq.example.com*.

Procedimiento

1. Instale IBM MQ en *sun* y *earth*.

   Si desea más información, consulte Instalación del servidor de IBM MQ en Windows .

   a) En *sun* y *earth*, inicie la sesión como administrador del dominio, *wmq*\Administrator.

   b) Ejecute el mandato **Setup** en el soporte de instalación de IBM MQ for Windows.

      Se iniciará la aplicación Launchpad de IBM MQ.

   c) Pulse **Requisitos de software** para comprobar si el software de requisitos previos está instalado.

   d) Pulse **Configuración de red** > **No**.

      Puede configurar un ID de usuario de domino o no para esta instalación. El ID de usuario que se ha creado es un ID de usuario local de dominio.

   e) Pulse **Instalación de IBM MQ**, seleccione un idioma de instalación y pulse Iniciar instalador de IBM MQ.

   f) Confirme el acuerdo de licencia y pulse **Siguiente** > **Siguiente** > **Instalar** para aceptar la configuración predeterminada. Espere hasta que se complete la instalación y pulse **Finalizar**.

      Si desea cambiar el nombre de la instalación, instale diferentes componentes, configure un directorio diferente para datos y registros del gestor de colas o realice la instalación en otro directorio, pulse **Personalizada** en vez de **Típica**.

      Se instala IBM MQ y el instalador inicia el Prepare IBM MQ Wizard.

   La instalación de IBM MQ for Windows configura un grupo local de dominio `mqm` y un grupo de dominio `Domain mqm`. Hace que `Domain mqm` sea miembro de `mqm`. Los controladores de dominio subsiguientes en el mismo dominio comparten los grupos `mqm` y `Domain mqm`.
2. En *earth* y *sun*, ejecute el Prepare IBM MQ Wizard.

Si desea más información, consulte Configuración de IBM MQ con el Prepare IBM MQ Wizard.

a) El instalador de IBM MQ ejecuta el Prepare IBM MQ Wizard automáticamente.

Para iniciar el asistente manualmente, busque el acceso directo al Prepare IBM MQ Wizard en la carpeta **Inicio** > **Todos los programas** > **IBM MQ**. Seleccione el acceso directo que corresponda a la instalación de IBM MQ en una configuración de varias instalaciones.

b) Pulse **Siguiente** y deje **No** pulsado en respuesta a la pregunta "Identificar si hay un controlador de dominio Windows 2000 o posterior en la red"[1].

c) En la página final del asistente, seleccione o borre los recuadros de selección tal como sea necesario y pulse **Finalizar**.

El Prepare IBM MQ Wizard crea un usuario local de dominio `MUSR_MQADMIN` en el primer controlador de dominio y otro usuario local de dominio `MUSR_MQADMIN1` en el segundo controlador de dominio. El asistente crea el servicio de IBM MQ en cada controlador, con `MUSR_MQADMIN` o `MUSR_MQADMIN1` como usuario que inicia la sesión en el servicio.

3. Defina un usuario que tenga permiso para crear un gestor de colas.

El usuario debe tener el derecho a iniciar la sesión localmente y ser miembro del grupo `mqm` local del dominio. En controladores de dominio, los usuarios de dominio no tienen derecho a iniciar sesión localmente, pero los administradores sí. De forma predeterminada, ningún usuario tiene ambos de estos atributos. En esta tarea, añada administradores de dominio al grupo `mqm` local del dominio.

a) Abra **Gestor de servidores** > **Roles** > **Servicios de dominio de Active Directory** > ***wmq.example.com*** > **Usuarios**.

b) Pulse con el botón derecho del ratón en **Administradores de dominio** > **Añadir a un grupo ...** y escriba `mqm` ; pulse **Comprobar nombres** > **Aceptar** > **Aceptar**

Resultados

1. Compruebe que el Prepare IBM MQ Wizard ha creado el usuario de dominio, `MUSR_MQADMIN`:
   a. Abra **Gestor de servidores** > **Roles** > **Servicios de dominio de Active Directory** > ***wmq.example.com*** > **Usuarios**.
   b. Pulse con el botón derecho del ratón en **MUSR_MQADMIN** > **Propiedades ...** > **Miembro de**y compruebe que es miembro de `Domain users` y `mqm`.
2. Compruebe si `MUSR_MQADMIN` tiene derecho a ejecutarse como servicio:
   a. PulsePulse **Iniciar** > **Ejecutar ...**, escriba el mandato **`secpol.msc`** y pulse **Aceptar**.
   b. Abrir **Configuración de seguridad** > **Políticas locales** > **Asignaciones de derechos de usuario**. En la lista de políticas, pulse con el botón derecho del ratón en **Iniciar sesión como servicio** > **Propiedades** y vea si `MUSR_MQADMIN` está listado con derecho a iniciar sesión como servicio. Pulse **Aceptar**.

Qué hacer a continuación

1. Realice la tarea, “Lectura y grabación de datos y archivos de registro autorizados por el grupo mqm local” en la página 566, para verificar que la instalación y configuración están funcionando correctamente.
2. Vuelva a la tarea, “Creación de un gestor de colas multiinstancia en controladores de dominio de Windows” en la página 551, para completar la tarea de configurar un gestor de colas multiinstancia en controladores de dominio.

Tareas relacionadas

Windows Adición de un segundo controlador de dominio de Windows a un dominio de ejemplo

[1] Puede configurar la instalación para el dominio. Puesto que todos los usuarios y grupos en un controlador de dominio tienen ámbito de dominio, no hay ninguna diferencia. Es más sencillo instalar IBM MQ como si no estuviera en el dominio.

Referencia relacionada

Derechos de usuario necesarios para un servicio IBM MQ Windows

Windows *Verificación del gestor de colas multiinstancia en Windows*

Utilice los programas de ejemplo **amqsghac**, **amqsphac** y **amqsmhac** para verificar la configuración de un gestor de colas multiinstancia. Este tema proporciona una configuración de ejemplo para verificar una configuración de gestor de colas multiinstancia en Windows Server 2003.

Los programas de ejemplo de alta disponibilidad utilizan la reconexión automática de cliente. Cuando falla el gestor de colas conectado, el cliente intenta volver a conectarse a un gestor de colas en el mismo grupo de gestores de colas. La descripción de los ejemplos, Programas de ejemplo de alta disponibilidad, muestra la reconexión de cliente mediante un gestor de colas de una sola instancia por razones de simplicidad. Se pueden utilizar los mismos ejemplos con gestores de colas multiinstancia para verificar una configuración de gestor de colas multiinstancia.

En este ejemplo se utiliza una configuración multiinstancia que se describe en el apartado “Creación de un gestor de colas multiinstancia en controladores de dominio de Windows” en la página 551. Utilice la configuración para verificar que el gestor de colas multiinstancia cambia a la instancia en espera. Detenga el gestor de colas con el mandato **endmqm** y utilice la opción de conmutación `-s`. Los programas cliente se reconectan a la nueva instancia del gestor de colas y continúan funcionando con la nueva instancia tras un ligero retardo.

El cliente está instalado en una imagen de VMware de 400 MB que ejecuta Windows 7 Service Pack 1. Por motivos de seguridad, está conectado en la misma red de sólo host VMware que los servidores de dominio que ejecutan el gestor de colas de varias instancias. Está compartiendo la carpeta `/MQHA`, que contiene la tabla de conexión de cliente, para simplificar la configuración.

Verificación de migración tras error utilizando IBM MQ Explorer

Antes de utilizar las aplicaciones de ejemplo para verificar las anomalías, ejecute IBM MQ Explorer en cada servidor. Añada ambas instancias del gestor de colas a cada explorador utilizando el asistente **Añadir gestor de colas remoto > Conectar directamente a un gestor de colas multiinstancia**. Asegúrese de que ambas instancias se estén ejecutando, permitiendo la espera. Cierre la ventana que ejecuta la imagen de VMware con la instancia activa, apagando el servidor virtualmente, o detenga la instancia activa, permitiendo la conmutación a la instancia en espera y que los clientes reconectables se reconecten.

Atención: Si apaga el servidor, asegúrese de que no sea el que aloja la carpeta `MQHA`.

Nota: Es posible que la opción **Permitir conmutación de una instancia en espera** no esté disponible en el diálogo **Detener gestor de colas**. La opción falta porque el gestor de colas se está ejecutando como un gestor de colas de una sola instancia. Deberá haberlo iniciado sin la opción **Permitir una instancia en espera**. Si la solicitud para detener el gestor de colas se rechaza, revise la ventana **Detalles**, posiblemente no habrá ninguna instancia en espera ejecutándose.

Verificación de anomalías mediante los programas de muestra

Elija un servidor para ejecutar la instancia activa

Es posible que haya elegido uno de los servidores para alojar el directorio o el sistema de archivos de `MQHA`. Si tiene previsto probar la migración tras error cerrando la ventana de VMware que ejecuta el servidor activo, asegúrese de que no es el que aloja `MQHA`.

En el servidor que ejecuta la instancia activa del gestor de colas

1. Modifique *ipaddr1* e *ipaddr2* y guarde los mandatos siguientes en `N:\hasample.tst`.

```
DEFINE QLOCAL(SOURCE) REPLACE
DEFINE QLOCAL(TARGET) REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN)  TRPTYPE(TCP) +
MCAUSER(' ') REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME(' ipaddr1 (1414), ipaddr2 (1414)') QMNAME(QM1) REPLACE
```

```
START CHANNEL(CHANNEL1)
DEFINE LISTENER(LISTENER.TCP) TRPTYPE(TCP) CONTROL(QMGR)
DISPLAY LISTENER(LISTENER.TCP) CONTROL
DISPLAY LSSTATUS(LISTENER.TCP) STATUS
```

Nota: Dejando el parámetro **MCAUSER** en blanco, el ID del usuario de cliente se enviará al servidor. El ID del usuario del cliente deberá tener los permisos correctos en los servidores. También puede establecerse el parámetro **MCAUSER** en el canal SVRCONN para el ID del usuario que ha configurado en el servidor.

2. Abra un indicador de mandatos con la vía de acceso `N:\` y ejecute el mandato:

```
runmqsc -m QM1 < hasample.tst
```

3. Verifique que el escucha se está ejecutando y tiene control sobre el gestor de colas examinando la salida del mandato **runmqsc**.

```
LISTENER(LISTENER.TCP)CONTROL(QMGR)
LISTENER(LISTENER.TCP)STATUS(RUNNING)
```

O, utilizando el IBM MQ Explorer que el escucha TCPIP está ejecutando y tiene `Control = Queue Manager`.

En el cliente

1. Correlacione el directorio compartido `C:\MQHA` en el servidor con `N:\` en el cliente.
2. Abra un indicador de mandatos con la vía de acceso `N:\`. Establezca la variable de entorno `MQCHLLIB` de manera que apunte a la tabla de definiciones de canales de clientes (CCDT) en el servidor:

```
SET MQCHLLIB=N:\data\QM1\@ipcc
```

3. En el indicador de mandatos introduzca los mandatos:

```
start amqsghac TARGET QM1
start amqsmhac -s SOURCE -t TARGET -m QM1
start amqsphac SOURCE QM1
```

Nota: Si tiene problemas, inicie las aplicaciones en un indicador de mandatos para que el código de razón se imprima en la consola o busque el archivo AMQERR01.LOG en la carpeta `N:\data\QM1\errors`.

En el servidor que ejecuta la instancia activa del gestor de colas

1. O bien:
   - Cierre la ventana que ejecuta la imagen de VMware con la instancia activa del servidor.
   - Mediante IBM MQ Explorer, detenga la instancia de gestor de colas activa, permitiendo el cambio a la instancia en espera e indicando a los clientes que se pueden volver a conectar que se reconecten.
2. Los tres clientes finalmente detectan que la conexión se ha interrumpido y vuelven a reconectarse. En esta configuración, si cierra la ventana de servidor, tarda unos siete minutos para que todas las conexiones vuelvan a restablecerse. Unas conexiones se restablecen antes que otras.

Resultados

```
N:\>amqsphac SOURCE QM1
Sample AMQSPHAC start
target queue is SOURCE
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

```
N:\>amqsmhac -s SOURCE -t TARGET -m QM1
Sample AMQSMHA0 start

17:05:25 : EVENT : Connection Reconnecting (Delay: 97ms)
17:05:48 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:53 : EVENT : Connection Reconnected
```

```
N:\>amqsghac TARGET QM1
Sample AMQSGHAC start
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 156ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

Windows *Proteger directorios y archivos de datos y registros compartidos del gestor de colas en Windows*

En este tema se describe cómo se puede proteger una ubicación compartida para archivos de datos y registros del gestor de colas utilizando un grupo de seguridad alternativo global. Puede compartir la ubicación entre dos instancias diferentes de un gestor de colas que se ejecuta en diferentes servidores.

Normalmente, no se configura una ubicación compartida para archivos de datos y registros del gestor de colas. Cuando instale IBM MQ for Windows, el programa de instalación creará un directorio de inicio que elija para los gestores de colas que están creados en dicho servidor. Protege los directorios con el grupo `mqm` local y configura un ID de usuario para que el servicio de IBM MQ acceda a los directorios.

Cuando protege una carpeta compartida con un grupo de seguridad, un usuario al que se le permite acceder a la carpeta debe tener las credenciales del grupo. Suponga que una carpeta en un servidor de archivos remotos está protegida mediante el grupo `mqm` local en un servidor denominado *`mars`*. Convierta al usuario que ejecuta los procesos del gestor de colas en miembro del grupo `mqm` local en *`mars`*. El usuario tiene las credenciales que coinciden con las credenciales de la carpeta en el servidor de archivos remoto. Mediante dichas credenciales, el gestor de colas puede acceder a los archivos de datos y registros en la carpeta. El usuario que ejecuta procesos del gestor de colas en un servidor diferente es miembro de un grupo `mqm` local diferente que no tiene credenciales coincidentes. Cuando el gestor de colas se ejecuta en un servidor distinto de *`mars`*, no puede acceder a los datos y a los archivos de registro que creó cuando se ejecutó en *`mars`*. Aunque convierta al usuario en usuario de dominio, tiene

diferentes credenciales porque debe adquirir las credenciales del grupo `mqm` local en *`mars`* y no puede hacerlo desde otro servidor.

Proporcionar al gestor de colas un grupo de seguridad alternativo global resuelve el problema; consulte la Figura 73 en la página 562. Proteja una carpeta remota con un grupo global. Pase el nombre del grupo global al gestor de colas cuando lo cree en *`mars`*. Pase el nombre del grupo global como grupo de seguridad alternativo utilizando el parámetro `-a[r]` en el mandato **`crtmqm`**. Si transfiere el gestor de colas para que se ejecute en otro servidor, el nombre del grupo de seguridad se transfiere con él. El nombre se transfiere en la stanza **AccessMode** del archivo `qm.ini` como `SecurityGroup`; por ejemplo:

```
AccessMode:
SecurityGroup=wmq\wmq
```

La stanza **AccessMode** de `qm.ini` también incluye `RemoveMQMAccess`; por ejemplo:

```
AccessMode:
RemoveMQMAccess=true|false
```

Si se especifica este atributo con el valor `true`, y también se ha facilitado un grupo de acceso, el grupo mqm local no recibe acceso a los archivos de datos del gestor de colas.

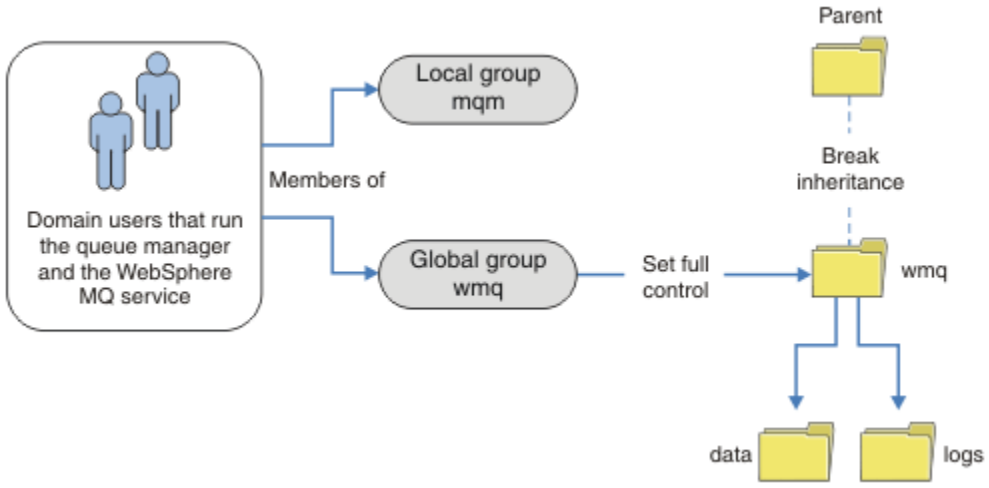


Figura 73. Protección de datos y registros del gestor de colas utilizando un grupo de seguridad global alternativo (1)

Para el ID de usuario con el que los procesos del gestor de colas van a realizar la ejecución para tener las credenciales coincidentes del grupo de seguridad global, el ID de usuario también debe tener ámbito global. No puede convertir un grupo o principal local en miembro de un grupo global. En la Figura 73 en la página 562, los usuarios que ejecutan los procesos del gestor de colas aparecen como usuarios de dominio.

Si está desplegando muchos servidores de IBM MQ, la agrupación de usuarios en la Figura 73 en la página 562 no es adecuada. Necesitará repetir el proceso de añadir usuarios a grupos locales para cada servidor de IBM MQ. En su lugar, cree un grupo global de `Domain mqm` en el controlador de dominio y haga que los usuarios que ejecutan IBM MQ sean miembros del grupo `Domain mqm`; consulte Figura 74 en la página 563. Cuando instala IBM MQ como una instalación de dominio, Prepare IBM MQ Wizard convierte automáticamente al grupo `Domain mqm` en miembro del grupo `mqm` local. Los mismos usuarios están en los grupos globales `Domain mqm` y `wmq`.

Consejo: Los mismos usuarios pueden ejecutar IBM MQ en diferentes servidores, pero en un servidor individual debe tener diferentes usuarios para ejecutar IBM MQ como servicio y ejecutarlo de forma

interactiva. También debe tener usuarios diferentes para cada instalación en un servidor. Por lo tanto, suele contener, `Domain mqm` contiene un número de usuarios.

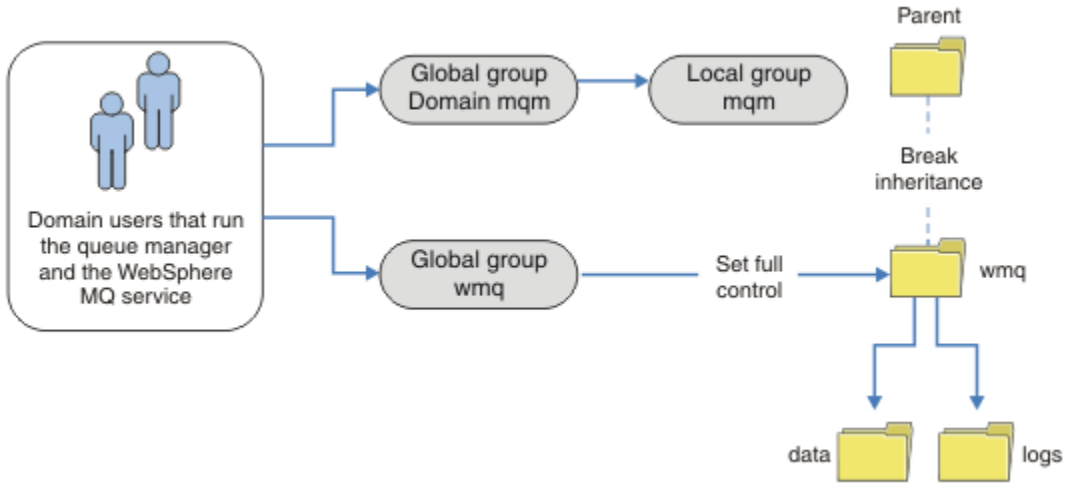


Figura 74. Protección de datos y registros del gestor de colas utilizando un grupo de seguridad global alternativo (2)

La organización en la Figura 74 en la página 563 es innecesariamente complicada tal como se presenta. La disposición tiene dos grupos globales con miembros idénticos. Puede simplificar la organización y definir únicamente un grupo global; consulte Figura 75 en la página 563.

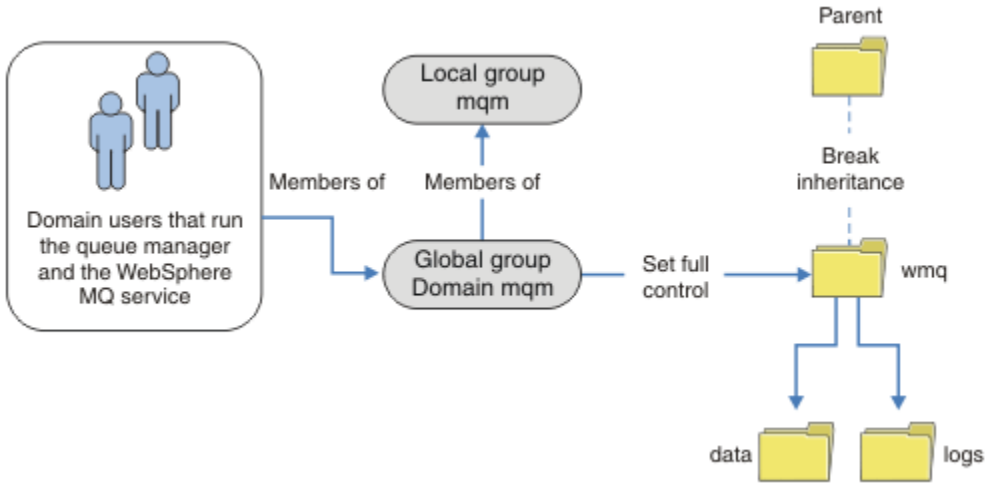


Figura 75. Protección de datos y registros del gestor de colas utilizando un grupo de seguridad global alternativo (3)

O bien, puede que necesite un grado más refinado de control de acceso, con diferentes gestores de colas restringidos para poder acceder a diferentes carpetas; consulte Figura 76 en la página 564. En la Figura 76 en la página 564, se definen dos grupos de usuarios de dominio, en grupos globales separados para proteger diferentes archivos de registros y datos del gestor de colas. Se muestran dos grupos `mqm` locales diferentes, que deben estar en distintos servidores de IBM MQ. En este ejemplo, los gestores de colas se dividen en dos conjuntos, con diferentes usuarios asignados a dos conjuntos. Los dos conjuntos pueden ser gestores de cola de prueba y de producción. Los grupos de seguridad alternativos se denominan `wmq1` y `wmq2`. Debe añadir los grupos globales `wmq1` y `wmq2` manualmente a los gestores de colas correctos según si están en el departamento de prueba o de producción. La configuración no puede aprovechar que

la instalación de IBM MQ propaga `Domain mqm` al grupo `mqm` local como en Figura 75 en la página 563, porque hay dos grupos de usuarios.

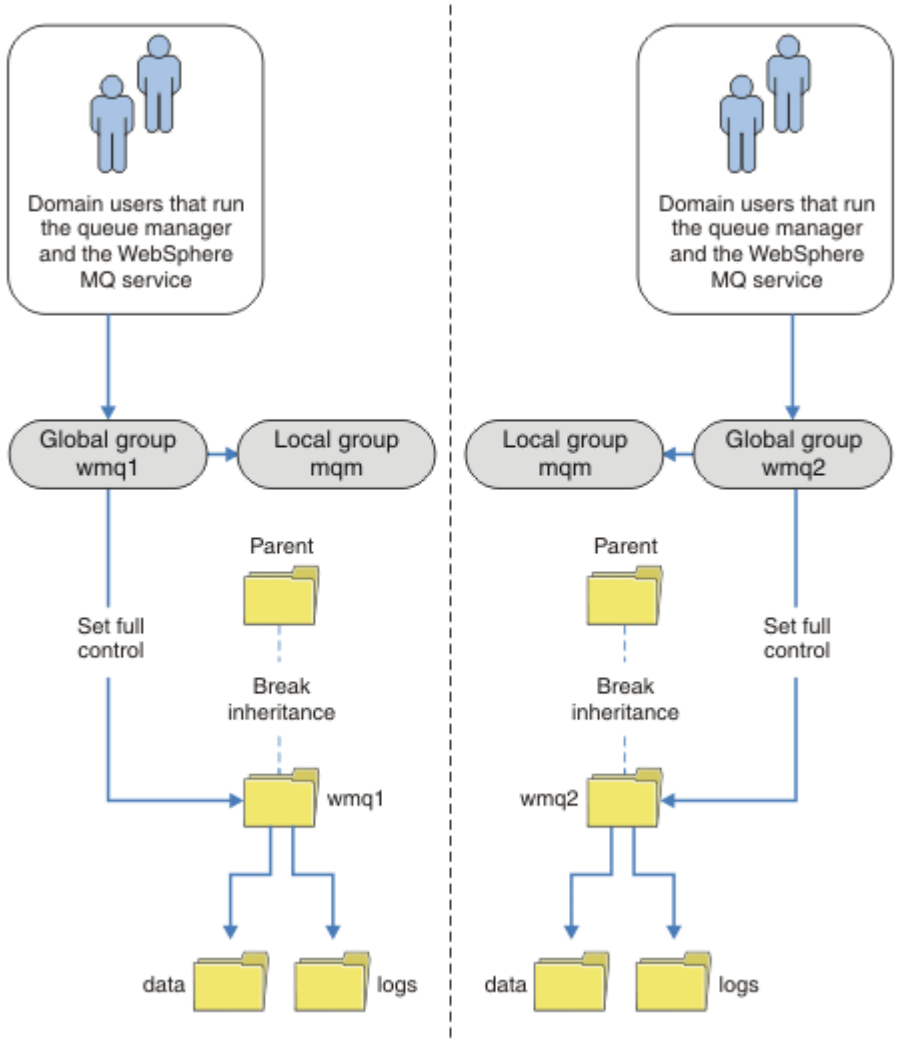


Figura 76. Protección de datos y registros del gestor de colas utilizando un principal de seguridad global alternativo (4)

Un método alternativo para dividir dos departamentos sería colocarlos en dos dominios de Windows. En este caso, podría volver a utilizar el método más simple que aparece en la Figura 75 en la página 563.

Windows *Proteger directorios y archivos de datos y registros del gestor de colas no compartidos en Windows*

En este tema se describe cómo proteger una ubicación alternativa para archivos de datos y registros del gestor de colas, utilizando el grupo local `mqm` como grupo de seguridad alternativo.

En general, no configure una ubicación alternativa para archivos de datos y registros del gestor de colas. Cuando instale IBM MQ for Windows, el programa de instalación creará un directorio de inicio de su elección para los gestores de colas que se hayan creado. Protege los directorios con el grupo `mqm` local y configura un ID de usuario para que el servicio de IBM MQ acceda a los directorios.

Dos ejemplos demuestran cómo configurar el control de acceso para IBM MQ. Los ejemplos muestran cómo crear un gestor de colas con los datos y registros en directorios que no están en las vías de acceso de los datos y registros creados por la instalación. En el primer ejemplo, "Lectura y grabación de datos y archivos de registro autorizados por el grupo mqm local" en la página 566, permite acceso a los directorios de colas y registros dándole autorización mediante el grupo `mqm` local. El segundo ejemplo,

"Leer y grabar archivos de datos y de registro autorizados por un grupo de seguridad local alternativo" en la página 569, difiere en el sentido de que el acceso a los directorios está autorizado mediante un grupo de seguridad alternativo. Cuando se accede a los directorios mediante un gestor de colas que ejecuta únicamente un servidor, proteger los archivos de datos y registros mediante el grupo de seguridad alternativo le proporciona la posibilidad de proteger diferentes gestores de colas con diferentes grupos o principales locales. Cuando un gestor de colas que se ejecuta en diferentes servidores, por ejemplo un gestor de colas multiinstancia, accede a los directorios, proteger los archivos de datos y registros mediante el grupo de seguridad alternativo es la única opción; consulte "Proteger directorios y archivos de datos y registros compartidos del gestor de colas en Windows" en la página 561.

La configuración de permisos de seguridad de archivos de datos y registros no es una tarea común en Windows. Cuando instale IBM MQ for Windows, especifique directorios para datos y registros del gestor de colas o acepte los directorios predeterminados. El programa de instalación protege automáticamente estos directorios con el grupo `mqm` local, otorgándole permiso de control completo. El proceso de instalación se asegura de que el ID de usuario que ejecuta gestores de colas sea miembro del grupo `mqm` local. Puede modificar los demás permisos de acceso sobre los directorios para cumplir los requisitos de acceso.

Si mueve el directorio de archivos de datos y registros a nuevas ubicaciones, debe configurar la seguridad de las nuevas ubicaciones. Puede cambiar la ubicación de los directorios si efectúa una copia de seguridad de un gestor de colas y lo restaura en otro sistema o si cambia el gestor de colas para que sea un gestor de colas multiinstancia. Tiene dos formas de proteger los directorios de datos y registros del gestor de colas a su nueva ubicación. Puede proteger los directorios limitando el acceso al grupo `mqm` local o puede limitar el acceso a cualquier grupo de seguridad que elija.

Proteger los directorios utilizando el grupo `mqm` local requiere un número mínimo de pasos. Establezca los permisos sobre los directorios de datos y registros que permitan al grupo `mqm` local tener un control completo. Una alternativa habitual es copiar el conjunto existente de permisos, eliminando la herencia del padre. A continuación, puede eliminar o restringir los permisos de otros principales.

Si ejecuta el gestor de colas bajo un ID de usuario diferente al servicio configurado por el Asistente de preparación de IBM MQ, dicho ID de usuario debe ser miembro del grupo local `mqm`. La tarea, "Lectura y grabación de datos y archivos de registro autorizados por el grupo mqm local" en la página 566, le guiará por diferentes pasos.

También puede proteger los archivos de datos y registros del gestor de colas utilizando un grupo de seguridad alternativo. El proceso de proteger los archivos de registro y datos del gestor de colas con el grupo de seguridad alternativo comprende diversas tareas que hacen referencia a Figura 77 en la página 565. El grupo local, `wmq`, es un ejemplo de un grupo de seguridad alternativo.

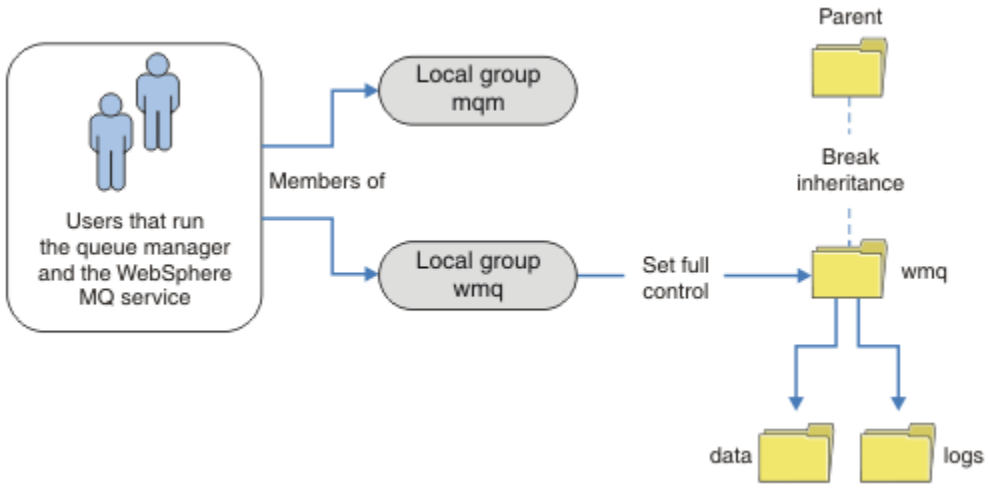


Figura 77. Protección de datos y registros del gestor de colas utilizando un grupo de seguridad local alternativo, `wmq`

1. Cree directorios separados para los datos y registros del gestor de colas, un directorio común o un directorio padre común.
2. Copie el conjunto existente de permisos heredados para los directorios o directorio padre y los modifica según sus necesidades.
3. Proteja los directorios que van a contener el gestor de colas y los registros otorgando al grupo alternativo, `wmq`, permiso de control completo a los directorios.
4. Otorgue a todos los ID de usuario que ejecutan procesos del gestor de colas las credenciales o el grupo o principal de seguridad alternativo:
   a. Si define un usuario como principal de seguridad alternativo, el usuario debe ser el mismo usuario bajo el que se va a ejecutar el gestor de colas. El usuario debe ser miembro del grupo `mqm` local.
   b. Si define un grupo local como grupo de seguridad alternativo, añada al usuario bajo el que se va a ejecutar el gestor de colas al grupo alternativo. El usuario también debe ser miembro del grupo `mqm` local.
   c. Si define un grupo global como grupo de seguridad alternativo, a continuación consulte "Proteger directorios y archivos de datos y registros compartidos del gestor de colas en Windows" en la página 561.
5. Cree el gestor de colas especificando el grupo o principal de seguridad alternativo en el mandato **`crtmqm`**, con el parámetro `-a`.

Windows *Lectura y grabación de datos y archivos de registro autorizados por el grupo mqm local*

La tarea ilustra cómo crear un gestor de colas con sus archivos de datos y registros almacenados en cualquier directorio que elija. El acceso a los archivos está protegido por el grupo `mqm` local. El directorio no es compartido.

Antes de empezar

1. Instale IBM MQ for Windows como la instalación principal.
2. Ejecute Prepare IBM MQ Wizard.

   Si desea más información, consulte Configuración de IBM MQ con el Prepare IBM MQ Wizard.

   Para esta tarea, configure la instalación para ejecutarla con un ID de usuario local o un ID de usuario de dominio. Eventualmente, para completar todas las tareas del apartado "Dominios de Windows y gestores de colas multiinstancia" en la página 534, la instalación debe configurarse para un dominio.
3. Inicie la sesión con derechos de administrador para realizar la primera parte de la tarea.

Acerca de esta tarea

Esta tarea es una de un conjunto de tareas relacionadas que ilustran cómo acceder a los datos del gestor de colas y a los archivos de registro. Las tareas muestran cómo crear un gestor de colas con autorización para leer y grabar archivos de datos y registros que están almacenados en un directorio que elija. Acompañan a la tarea, "Dominios de Windows y gestores de colas multiinstancia" en la página 534.

En Windows, puede crear las vías de acceso de datos y registros predeterminadas para IBM MQ for Windows en cualquier directorio que elija. El asistente de instalación y configuración otorga automáticamente al grupo `mqm` local y al ID de usuario que está ejecutando los procesos del gestor de colas, acceso a los directorios. Si crea un gestor de colas especificando directorios diferentes para archivos de datos y registros del gestor de colas, debe configurar permiso de control completo sobre los directorios.

En este ejemplo, le otorga al gestor de colas control completo sobre sus datos y archivos de registro, proporcionando al grupo local `mqm` permiso para el directorio *`c:\wmq`*.

El mandato **`crtmqm`** crea un gestor de colas que se inicia automáticamente cuando se inicia la estación de trabajo utilizando el servicio de IBM MQ.

La tarea es ilustrativa; utiliza valores específicos que puede cambiar. Los valores que puede cambiar están en cursiva. Al final de la tarea, siga las instrucciones para eliminar todos los cambios que haya efectuado.

Procedimiento

1. Abra un indicador de mandatos.
2. Escriba el mandato:

```
md c:\wmq\data, c:\wmq\logs
```

3. Establezca los permisos de los directorios para permitir el acceso de lectura y escritura del grupo `mqm` local.

```
cacls c:\wmq/T /E /G mqm:F
```

   La respuesta del sistema:

```
processed dir: c:\wmq
processed dir: c:\wmq\data
processed dir: c:\wmq\logs
```

4. Opcional: Cambie a un ID de usuario que sea miembro del grupo `mqm` local.

   Puede continuar como administrador, pero para una configuración de producción realista, continúe con un ID de usuario con más derechos restringidos. El ID de usuario debe ser como mínimo un miembro del grupo `mqm` local.

   Si la instalación de IBM MQ está configurada como parte de un dominio, haga que el ID de usuario sea miembro del grupo `Domain mqm`. El asistente de "preparación de IBM MQ" hace que el grupo global `Domain mqm` sea miembro del grupo `mqm` local, por lo que no es necesario que el ID de usuario sea directamente un miembro del grupo `mqm` local.

5. Cree el gestor de colas.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md c:\wmq\data -ld c:\wmq\logs QMGR
```

   La respuesta del sistema:

```
IBM MQ queue manager created.
Directory 'c:\wmq\data\QMGR' created.
The queue manager is associated with installation '1'
Creating or replacing default objects for queue manager 'QMGR'
Default objects statistics : 74 created. 0 replaced.
Completing setup.
Setup completed.
```

6. Compruebe que los directorios creados por el gestor de colas se encuentran en el directorio *c:\wmq*.

```
dir c:\wmq/D /B /S
```

7. Compruebe que los archivos tienen permiso de lectura y escritura o de control completo para el grupo `mqm` local.

```
cacls c:\wmq\*.*
```

Qué hacer a continuación

Pruebe el gestor de colas transfiriendo y obteniendo un mensaje de una cola.

1. Inicie el gestor de colas.

   ```
   strmqm QMGR
   ```

   La respuesta del sistema:

   ```
   IBM MQ queue manager 'QMGR' starting.
   The queue manager is associated with installation '1'.
   5 log records accessed on queue manager 'QMGR' during the log
   replay phase.
   Log replay for queue manager 'QMGR' complete.
   Transaction manager state recovered for queue manager 'QMGR'.
   IBM MQ queue manager 'QMGR' started using V7.1.0.0.
   ```

2. Cree una cola de prueba.

   ```
   echo define qlocal(QTEST) | runmqsc QMGR
   ```

   La respuesta del sistema:

   ```
   5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
   Starting MQSC for queue manager QMGR.


   1 : define qlocal(QTEST)
   AMQ8006: IBM MQ queue created.
   One MQSC command read.
   No commands have a syntax error.
   All valid MQSC commands were processed.
   ```

3. Transfiera un mensaje de prueba utilizando el programa de ejemplo **amqsput**.

   ```
   echo 'A test message' | amqsput QTEST QMGR
   ```

   La respuesta del sistema:

   ```
   Sample AMQSPUT0 start
   target queue is QTEST
   Sample AMQSPUT0 end
   ```

4. Obtenga el mensaje de prueba utilizando el programa de ejemplo **amqsget**.

   ```
   amqsget QTEST QMGR
   ```

   La respuesta del sistema:

   ```
   Sample AMQSGET0 start
   message A test message
   Wait 15 seconds ...
   no more messages
   Sample AMQSGET0 end
   ```

5. Detenga el gestor de colas.

```
endmqm -i QMGR
```

   La respuesta del sistema:

```
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ended.
```

6. Suprima el gestor de colas.

```
dltmqm QMGR
```

   La respuesta del sistema:

```
IBM MQ queue manager 'QMGR' deleted.
```

7. Suprima los directorios que ha creado.

   Consejo: Añada la opción /Q a los mandatos para impedir que el mandato solicite la supresión de cada archivo o directorio.

```
del /F /S C:\wmq\*.*
rmdir /S C:\wmq
```

Conceptos relacionados

“Dominios de Windows y gestores de colas multiinstancia” en la página 534
Un gestor de colas multiinstancia en Windows requiere que se compartan sus datos y registros. El compartimiento debe ser accesible para todas las instancias del gestor de colas que se ejecutan en diferentes servidores o estaciones de trabajo. Configure los gestores de colas y compártalos como parte de un dominio de Windows. El gestor de colas se puede ejecutar en una estación de trabajo o servidor de dominio o en el controlador de dominio.

Tareas relacionadas

Windows Leer y grabar archivos de datos y de registro autorizados por un grupo de seguridad local alternativo
Esta tarea muestra cómo utilizar el distintivo -a en el mandato **crtmqm**. El distintivo proporciona al gestor de colas un grupo de seguridad local alternativo para darle acceso a sus archivos de registros y datos

“Leer y grabar archivos de datos y de registro compartidos autorizados por un grupo de seguridad global alternativo” en la página 548
“Creación de gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio en Windows” en la página 536

Windows *Leer y grabar archivos de datos y de registro autorizados por un grupo de seguridad local alternativo*
Esta tarea muestra cómo utilizar el distintivo -a en el mandato **crtmqm**. El distintivo proporciona al gestor de colas un grupo de seguridad local alternativo para darle acceso a sus archivos de registros y datos

Antes de empezar

1. Instale IBM MQ for Windows como la instalación principal.
2. Ejecute Prepare IBM MQ Wizard.

   Si desea más información, consulte Configuración de IBM MQ con el Prepare IBM MQ Wizard.

Para esta tarea, configure la instalación para ejecutarla con un ID de usuario local o un ID de usuario de dominio. Eventualmente, para completar todas las tareas del apartado "Dominios de Windows y gestores de colas multiinstancia" en la página 534, la instalación debe configurarse para un dominio.

3. Inicie la sesión con derechos de administrador para realizar la primera parte de la tarea.

Acerca de esta tarea

Esta tarea es una de un conjunto de tareas relacionadas que ilustran cómo acceder a los datos del gestor de colas y a los archivos de registro. Las tareas muestran cómo crear un gestor de colas con autorización para leer y grabar archivos de datos y registros que están almacenados en un directorio que elija. Acompañan a la tarea, "Dominios de Windows y gestores de colas multiinstancia" en la página 534.

En Windows, puede crear las vías de acceso de datos y registros predeterminadas para IBM MQ for Windows en cualquier directorio que elija. El asistente de instalación y configuración otorga automáticamente al grupo `mqm` local y al ID de usuario que está ejecutando los procesos del gestor de colas, acceso a los directorios. Si crea un gestor de colas especificando directorios diferentes para archivos de datos y registros del gestor de colas, debe configurar permiso de control completo sobre los directorios.

En este ejemplo, proporcione al gestor de registros un grupo local de seguridad alternativo que tenga autorización de control completa sobre los directorios. El grupo de seguridad alternativo proporciona permiso al gestor de colas para gestionar archivos en el directorio. La finalidad primaria del grupo de seguridad alternativo es autorizar un grupo global de seguridad alternativo. Utilice un grupo global de seguridad alternativo para configurar un gestor de colas multiinstancia. En este ejemplo, configure un grupo local para que se familiarice con el uso de un grupo de seguridad alternativo sin instalar IBM MQ en un dominio. Es poco habitual configurar un grupo local como grupo de seguridad alternativo.

La tarea es ilustrativa; utiliza valores específicos que puede cambiar. Los valores que puede cambiar están en cursiva. Al final de la tarea, siga las instrucciones para eliminar todos los cambios que haya efectuado.

Procedimiento

1. Configure un grupo de seguridad alternativo.

   El grupo de seguridad alternativo suele ser un grupo de dominio. En el ejemplo, cree un gestor de colas que utilice un grupo de seguridad local alternativo. Con un grupo de seguridad local alternativo, puede realizar la tarea con una instalación de IBM MQ que no forme parte de un dominio.

   a) Ejecute el mandato **`lusrmgr.msc`** para abrir la ventana Usuarios locales y grupos.

   b) Pulse con el botón derecho **Grupos** > **Nuevo Grupo ...**

   c) En el campo **Nombre de grupo** , escriba *`altmqm`* y pulse **Crear** > **Cerrar**.

   d) Identifique el ID de usuario que ejecuta el servicio de IBM MQ.

      i) Pulse **Inicio** > **Ejecutar...**, escriba `services.msc` y pulse **Aceptar**.

      ii) Pulse el servicio de IBM MQ en la lista de servicios y pulse la pestaña Iniciar sesión.

      iii) Recuerde el ID de usuario y cierre el Explorador de servicios.

   e) Añada el ID de usuario que ejecuta el servicio IBM MQ al grupo *`altmqm`*. Además, añada el ID de usuario con el que ha iniciado la sesión para crear un gestor de colas y ejecútelo interactivamente.

      Windows comprueba la autorización del gestor de colas para acceder a los directorios de datos y registros comprobando la autorización del ID de usuario que está ejecutando procesos del gestor de colas. El ID de usuario debe ser miembro, directa o indirectamente, a través de un grupo global, del grupo *`altmqm`* que ha autorizado los directorios.
      Si ha instalado IBM MQ como parte de un dominio y va a realizar las tareas en "Creación de gestor de colas multiinstancia en estaciones de trabajo o servidores de dominio en Windows" en la página 536, los ID de usuario de dominio creados en "Creación de un dominio de Active Directory y DNS en Windows" en la página 539 son *`wmquser1`* y *`wmquser2`*.

Si no ha instalado el gestor de colas como parte de un dominio, el ID de usuario local predeterminado que ejecuta el servicio IBM MQ es MUSR_MQADMIN. Si tiene la intención de realizar las tareas sin autorización de administrador, cree un usuario que sea miembro del grupo mqm local.

Siga estos pasos para añadir *wmquser1* y *wmquser2* a *altmqm*. Si la configuración es diferente, sustituya los nombres por los ID de usuario y grupo.

i) En la lista de grupos, pulse con el botón derecho del ratón en **altmqm** > **Propiedades** > **Agregar...**.

ii) En la ventana Seleccionar usuarios, equipos o grupos, escriba *wmquser1 ; wmquser2* y pulse **Comprobar nombres**.

iii) Escriba el nombre y la contraseña de un administrador de dominio en la ventana Seguridad de Windows y luego pulse **Aceptar** > **Aceptar** > **Aplicar** > **Aceptar**.

2. Abra un indicador de mandatos.
3. Reinicie el servicio IBM MQ.

   Debe reiniciar el servicio para que el ID de usuario bajo el que se ejecuta adquiera las credenciales de seguridad adicionales que ha configurado para ello.

   Escriba los mandatos:

   ```
   endmqsvc
   strmqsvc
   ```

   Las respuestas del sistema:

   ```
   5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
   The MQ service for installation 'Installation1' ended successfully.
   ```

   Y:

   ```
   5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
   The MQ service for installation 'Installation1' started successfully.
   ```

4. Escriba el mandato:

   ```
   md c:\wmq\data, c:\wmq\logs
   ```

5. Establezca los permisos en los directorios para permitir al usuario local *user* acceso de lectura y escritura.

   ```
   cacls c:\wmq/T /E /G altmqm:F
   ```

   La respuesta del sistema:

   ```
   processed dir: c:\wmq
   processed dir: c:\wmq\data
   processed dir: c:\wmq\logs
   ```

6. Opcional: Cambie a un ID de usuario que sea miembro del grupo mqm local.

   Puede continuar como administrador, pero para una configuración de producción realista, continúe con un ID de usuario con más derechos restringidos. El ID de usuario debe ser como mínimo un miembro del grupo mqm local.

   Si la instalación de IBM MQ está configurada como parte de un dominio, haga que el ID de usuario sea miembro del grupo Domain mqm. El asistente de "preparación de IBM MQ" hace que el grupo global Domain mqm sea miembro del grupo mqm local, por lo que no es necesario que el ID de usuario sea directamente un miembro del grupo mqm local.

7. Cree el gestor de colas.

```
crtmqm -a altmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md c:\wmq\data -ld c:\wmq\logs QMGR
```

   La respuesta del sistema:

```
IBM MQ queue manager created.
Directory 'c:\wmq1\data\QMGR' created.
The queue manager is associated with installation '1'
Creating or replacing default objects for queue manager 'QMGR'
Default objects statistics : 74 created. 0 replaced.
Completing setup.
Setup completed.
```

8. Compruebe que los directorios creados por el gestor de colas se encuentran en el directorio *c:\wmq*.

```
dir c:\wmq/D /B /S
```

9. Compruebe que los archivos tienen permiso de lectura y escritura o de control completo para el grupo mqm local.

```
cacls c:\wmq\*.*
```

Qué hacer a continuación

Pruebe el gestor de colas transfiriendo y obteniendo un mensaje de una cola.

1. Inicie el gestor de colas.

```
strmqm QMGR
```

   La respuesta del sistema:

```
IBM MQ queue manager 'QMGR' starting.
The queue manager is associated with installation '1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
IBM MQ queue manager 'QMGR' started using V7.1.0.0.
```

2. Cree una cola de prueba.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

   La respuesta del sistema:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
Starting MQSC for queue manager QMGR.


1 : define qlocal(QTEST)
AMQ8006: IBM MQ queue created.
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

3. Transfiera un mensaje de prueba utilizando el programa de ejemplo **amqsput**.

```
echo 'A test message' | amqsput QTEST QMGR
```

   La respuesta del sistema:

```
Sample AMQSPUT0 start
target queue is QTEST
Sample AMQSPUT0 end
```

4. Obtenga el mensaje de prueba utilizando el programa de ejemplo **amqsget**.

```
amqsget QTEST QMGR
```

   La respuesta del sistema:

```
Sample AMQSGET0 start
message A test message
Wait 15 seconds ...
no more messages
Sample AMQSGET0 end
```

5. Detenga el gestor de colas.

```
endmqm -i QMGR
```

   La respuesta del sistema:

```
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ended.
```

6. Suprima el gestor de colas.

```
dltmqm QMGR
```

   La respuesta del sistema:

```
IBM MQ queue manager 'QMGR' deleted.
```

7. Suprima los directorios que ha creado.

   Consejo: Añada la opción `/Q` a los mandatos para impedir que el mandato solicite la supresión de cada archivo o directorio.

```
del /F /S C:\wmq\*.*
rmdir /S C:\wmq
```

Tareas relacionadas

Windows Lectura y grabación de datos y archivos de registro autorizados por el grupo mqm local

La tarea ilustra cómo crear un gestor de colas con sus archivos de datos y registros almacenados en cualquier directorio que elija. El acceso a los archivos está protegido por el grupo `mqm` local. El directorio no es compartido.

Linux *Creación de un gestor de colas multiinstancia en Linux*

Un ejemplo que muestra cómo configurar un gestor de colas de varias instancias en Linux. La configuración es reducida para mostrar los conceptos involucrados. El ejemplo se basa en Linux Red Hat Enterprise 5. Los pasos difieren en otras plataformas de UNIX.

Acerca de esta tarea

El ejemplo se configura en un sistema portátil de 2 GHz con 3 GB de RAM ejecutando Windows 7 Service Pack 1. Dos máquinas virtuales VMware, Server1 y Server2, ejecutan Linux Red Hat Enterprise 5 en imágenes de 640 MB. En Server1 se aloja el sistema de archivos de red (NFS), los registros del gestor de colas y una instancia HA. No es una práctica habitual que en el servidor NFS también se aloje una de las instancias del gestor de colas; esto es para simplificar el ejemplo. La máquina Server2 monta los registros del gestor de colas de Server1 con una instancia que se encuentra en espera. Un cliente MQI de WebSphere MQ está instalado en una imagen VMware de 400 MB adicional que ejecuta Windows 7 Service Pack 1 y ejecuta las aplicaciones de ejemplo de alta disponibilidad. Todas las máquinas virtuales están configuradas como parte de una red solo de host de VMware por razones de seguridad.

Nota: Debe colocar solo datos del gestor de colas en un servidor NFS. En el NFS, utilice las tres opciones siguientes con el mandato de montaje para proteger el sistema:

- **noexec**
  Con esta opción, no se pueden ejecutar archivos binarios en el NFS, lo que impide que un usuario remoto ejecute código no deseado en el sistema.
- **nosuid**
  Con esta opción, no se pueden utilizar los bits set-user-identifier y set-group-identifier, lo que impide que un usuario remoto obtenga mayores privilegios.
- **nodev**
  Con esta opción, no se pueden utilizar ni definir dispositivos especiales de bloque o caracteres, lo que impide a un usuario remoto salir de una cárcel chroot.

Procedimiento

1. Inicie una sesión como usuario root.
2. Lea Instalación de IBM MQ - visión general y siga el enlace adecuado para instalar IBM MQ, crear el usuario y el grupo mqm y definir `/var/mqm`.
3. Complete la tarea Verificar el comportamiento del sistema de archivos compartido para comprobar que el sistema de archivos da soporte a gestores de colas multiinstancia.
4. Para Server1, realice el paso siguiente:
   a. Cree directorios de registro y datos en una carpeta común, `/MQHA`, que se va a compartir. Por ejemplo:
      i) **mkdir** */MQHA*
      ii) **mkdir** */MQHA/logs*
      iii) **mkdir** */MQHA/qmgrs*
5. Para Server2, realice el paso siguiente:
   a. Cree la carpeta, `/MQHA`, para montar el sistema de archivos compartidos. Mantenga la misma vía de acceso que en Server1. Por ejemplo:
      i) **mkdir** */MQHA*
6. Asegúrese de que los directorios MQHA son propiedad del usuario y el grupo mqm, y que los permisos de acceso están definidos en `rwx` para el usuario y el grupo. Por ejemplo, **ls -al** muestra `drwxrwxr-x mqm mqm 4096 Nov 27 14:38 MQDATA`.
   a. **chown -R** `mqm:mqm` */MQHA*
   b. **chmod -R** `ug+rwx` */MQHA*

7. Cree el gestor de colas especificando el mandato siguiente: **crtmqm -ld** */MQHA/logs* **-md** */MQHA/qmgrs QM1*
8. Añadir[2] `/MQHA *(rw,sync,no_wdelay,fsid=0)` a `/etc/exports`
9. Para Server1, realice los pasos siguientes:
   a. Inicie el daemon NFS : */etc/init.d/* **nfs** `start`
   b. Copie los detalles de configuración del gestor de colas de Server1:

      ```
      dspmqinf -o command QM1
      ```

      y copie el resultado al portapapeles,

      ```
      addmqinf -s QueueManager
      -v Name=QM1
      -v Directory=QM1
      -v Prefix=/var/mqm
      -v DataPath=/MQHA/qmgrs/QM1
      ```

10. Para Server2, realice los pasos siguientes:
    a. Monte el sistema de archivos exportado `/MQHA` especificando el mandato siguiente: **mount -t** `nfs4` **-o** `hard,intr` *Server1:/ /MQHA*
    b. Pegue el mandato de configuración del gestor de colas en Server2,

       ```
       addmqinf -s QueueManager
       -v Name=QM1
       -v Directory=QM1
       -v Prefix=/var/mqm
       -v DataPath=/MQHA/qmgrs/QM1
       ```

11. Inicie las instancias del gestor de colas, en cualquier orden, con el parámetro-**x** : **strmqm** `-x` `QM1`.

    El mandato utilizado para iniciar las instancias del gestor de colas debe emitirse desde la misma instalación de IBM MQ que el mandato **addmqinf**. Para iniciar y detener el gestor colas de una instalación diferente, primero debe establecer la instalación asociada con el gestor de colas, mediante el mandato **setmqm**. Para obtener más información, consulte setmqm.

Linux *Verificación del gestor de colas multiinstancia en Linux*

Utilice los programas de ejemplo **amqsghac**, **amqsphac** y **amqsmhac** para verificar la configuración de un gestor de colas multiinstancia. Este tema proporciona una configuración de ejemplo para verificar una configuración de gestor de colas multiinstancia en Linux Red Hat Enterprise 5.

Los programas de ejemplo de alta disponibilidad utilizan la reconexión automática de cliente. Cuando falla el gestor de colas conectado, el cliente intenta volver a conectarse a un gestor de colas en el mismo grupo de gestores de colas. La descripción de los ejemplos, Programas de ejemplo de alta disponibilidad, muestra la reconexión de cliente mediante un gestor de colas de una sola instancia por razones de simplicidad. Se pueden utilizar los mismos ejemplos con gestores de colas multiinstancia para verificar una configuración de gestor de colas multiinstancia.

El ejemplo utiliza una configuración multiinstancia que se describe en el apartado “Creación de un gestor de colas multiinstancia en Linux” en la página 574. Utilice la configuración para verificar que el gestor de colas multiinstancia cambia a la instancia en espera. Detenga el gestor de colas con el mandato **endmqm** y utilice la opción de conmutación `-s`. Los programas cliente se reconectan a la nueva instancia del gestor de colas y continúan funcionando con la nueva instancia tras un ligero retardo.

En el ejemplo, el cliente se ejecuta en un sistema Windows 7 Service Pack 1. El sistema aloja dos servidores VMware Linux que están ejecutando el gestor de colas multiinstancia.

[2] El '*' permite que todas las máquinas que pueden alcanzarlo monten /MQHA para lectura/escritura. Restrinja el acceso en una máquina de producción.

Verificación de migración tras error utilizando IBM MQ Explorer

Antes de utilizar las aplicaciones de ejemplo para verificar las anomalías, ejecute IBM MQ Explorer en cada servidor. Añada ambas instancias del gestor de colas a cada explorador utilizando el asistente **Añadir gestor de colas remoto > Conectar directamente a un gestor de colas multiinstancia**. Asegúrese de que ambas instancias se estén ejecutando, permitiendo la espera. Cierre la ventana que ejecuta la imagen de VMware con la instancia activa, apagando el servidor virtualmente, o detenga la instancia activa, permitiendo el cambio a la instancia en espera.

Nota: Si apaga el servidor, asegúrese de que no es el que aloja `/MQHA`.

Nota: Es posible que la opción **Permitir conmutación de una instancia en espera** no esté disponible en el diálogo **Detener gestor de colas**. La opción falta porque el gestor de colas se está ejecutando como un gestor de colas de una sola instancia. Deberá haberlo iniciado sin la opción **Permitir una instancia en espera**. Si la solicitud para detener el gestor de colas se rechaza, revise la ventana **Detalles**, posiblemente no habrá ninguna instancia en espera ejecutándose.

Verificación de anomalías mediante los programas de muestra

Elija un servidor para ejecutar la instancia activa

Es posible que haya elegido uno de los servidores para alojar el directorio o el sistema de archivos de `MQHA`. Si tiene previsto probar la migración tras error cerrando la ventana de VMware que ejecuta el servidor activo, asegúrese de que no es el que aloja `MQHA`.

En el servidor que ejecuta la instancia activa del gestor de colas

Nota: Es conveniente ejecutar el canal SVRCONN con `MCAUSER` establecido en `mqm` para reducir el número de pasos de la configuración de ejemplo. Si se ha elegido otro ID de usuario y el sistema está configurado de forma diferente al del ejemplo, es posible que tenga algún problema de permisos para el acceso. No utilice `mqm` como un `MCAUSER` en un sistema expuesto; puede ser arriesgado para la seguridad.

1. Modifique *ipaddr1* e *ipaddr2* y guarde los mandatos siguientes en `/MQHA/hasamples.tst.`

```
DEFINE QLOCAL(SOURCE) REPLACE
DEFINE QLOCAL(TARGET) REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN)  TRPTYPE(TCP) +
MCAUSER('mqm') REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME(' ipaddr1 (1414), ipaddr2
(1414)') QMNAME(QM1) REPLACE
START  CHANNEL(CHANNEL1)
DEFINE LISTENER(LISTENER.TCP) TRPTYPE(TCP) CONTROL(QMGR)
DISPLAY LISTENER(LISTENER.TCP) CONTROL
START  LISTENER(LISTENER.TCP)
DISPLAY LSSTATUS(LISTENER.TCP) STATUS
```

2. Abra una ventana de terminal con la vía de acceso `/MQHA` y ejecute el mandato:

```
runmqsc -m QM1 < hasamples.tst
```

3. Verifique que el escucha se está ejecutando y tiene control sobre el gestor de colas examinando la salida del mandato **runmqsc**.

```
LISTENER(LISTENER.TCP)CONTROL(QMGR)
LISTENER(LISTENER.TCP)STATUS(RUNNING)
```

   O, utilizando el IBM MQ Explorer que el escucha TCPIP está ejecutando y tiene `Control = Queue Manager`.

En el cliente

1. Copie la tabla de conexión de cliente AMQCLCHL.TAB de `/MQHA/qmgrs/QM1.000/@ipcc` en el servidor en `C:\` en el cliente.

2. Abra un indicador de mandatos con la vía de acceso `C:\` y establezca la variable en entorno `MQCHLLIB` para que apunte a la tabla de definiciones de canales de clientes (CCDT).

```
SET MQCHLLIB=C:\
```

3. En el indicador de mandatos introduzca los mandatos:

```
start amqsghac TARGET QM1
start amqsmhac -s SOURCE -t TARGET -m QM1
start amqsphac SOURCE QM1
```

En el servidor que ejecuta la instancia activa del gestor de colas

1. O bien:
   - Cierre la ventana que ejecuta la imagen de VMware con la instancia activa del servidor.
   - Mediante IBM MQ Explorer, detenga la instancia de gestor de colas activa, permitiendo el cambio a la instancia en espera y ordenando a los clientes reconectables que se vuelvan a conectar.
2. Los tres clientes finalmente detectan que la conexión se ha interrumpido y vuelven a reconectarse. En esta configuración, si cierra la ventana de servidor, tarda unos siete minutos para que todas las conexiones vuelvan a restablecerse. Unas conexiones se restablecen antes que otras.

Resultados

```
N:\>amqsphac SOURCE QM1
Sample AMQSPHAC start
target queue is SOURCE
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

```
N:\>amqsmhac -s SOURCE -t TARGET -m QM1
Sample AMQSMHA0 start

17:05:25 : EVENT : Connection Reconnecting (Delay: 97ms)
17:05:48 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:53 : EVENT : Connection Reconnected
```

```
N:\>amqsghac TARGET QM1
Sample AMQSGHAC start
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 156ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

Linux AIX *Conversión de una única instancia en un gestor de colas de varias instancias en AIX and Linux*

Para convertir un gestor de colas de una sola instancia en un gestor de colas de varias instancias en AIX and Linux, debe mover los datos del gestor de colas a un directorio compartido y volver a configurar el gestor de colas en otros dos servidores.

Antes de empezar

Debe comprobar los requisitos previos para ejecutar un gestor de colas de varias instancias como parte de esta tarea.Para obtener una lista de entornos probados, consulte Declaración de prueba para sistemas de archivos de gestor de colas de varias instancias de IBM MQ. Otros entornos pueden funcionar; se proporciona una herramienta de prueba con IBM MQ para ayudarle a calificar otros entornos.

Debe tener tres servidores que ejecuten un gestor de colas multiinstancia. Un servidor tiene un sistema de archivos compartido para almacenar los datos y registros del gestor de colas. Los demás servidores ejecutan instancias activas y en espera del gestor de colas.

Acerca de esta tarea

Tiene un solo gestor de colas de instancias que desea convertir a un gestor de colas de varias instancias. La conversión del gestor de colas es en sí sencilla, pero debe hacer que otras tareas creen un entorno de producción totalmente automatizado.

Debe comprobar los requisitos previos para un gestor de colas de varias instancias, configurar el entorno y comprobarlo. Debe configurar un sistema da supervisión y de gestión para detectar si el gestor de colas de varias instancias ha fallado y se ha reiniciado automáticamente. A continuación, puede averiguar la causa del reinicio, repararlo y reiniciar el sistema en espera. También debe modificar aplicaciones o el modo en que las aplicaciones están conectadas al gestor de colas, de forma que puedan reanudar el proceso después de un reinicio del gestor de colas.

Procedimiento

1. Compruebe el sistema operativo en el que va a ejecutar el gestor de colas y el sistema de archivos en el que se almacenan los datos y los registros del gestor de colas. Compruebe si pueden ejecutar un gestor de colas de varias instancias.
   a) Consulte Declaración de prueba de sistemas de archivos del gestor de colas multiinstancia de IBM MQ. Consulte si la combinación de sistema operativo y sistema de archivos está probada y es capaz de ejecutar un gestor de colas de varias instancias.

      Un sistema de archivos debe proporcionar un bloqueo basado en arrendamiento para ejecutar gestores de colas de varias instancias. El bloqueo basado en arrendamiento es una característica reciente de algunos sistemas de archivos compartidos y en algunos casos se requieren arreglos. La declaración de soporte le proporciona la información básica.

   b) Ejecute **amqmfsck** para verificar si el sistema de archivos está configurado correctamente.

      A veces los sistemas de archivos se configuran con un rendimiento superior a través de integridad de datos. Es importante comprobar la configuración del sistema de archivos. Un informe negativo de la herramienta **amqmfsck** le indicará que los valores no son adecuados. Un resultado positivo es una indicación de que el sistema de archivos es correcto, pero el resultado no es una sentencia definitiva de que el sistema de archivos es correcto. Es una buena indicación.

   c) Ejecute la aplicación de comprobación de integridad que se proporciona en la nota técnica, Prueba de un sistema de archivos compartido para la compatibilidad con gestores de colas de varias instancias de IBM MQ.

      La aplicación de comprobación prueba si el gestor de colas se reinicia correctamente.
2. Configure que un usuario y un grupo puedan acceder a una unidad compartida en el sistema de archivos de red de cada servidor que esté ejecutando una instancia del gestor de colas.

   En AIX and Linux, `uid` y `gid` para `mqm` en `/etc/password` deben ser los mismos en cada sistema; consulte Creación de un gestor de colas de varias instancias en Linux .

3. Configure un directorio para la unidad compartida en el sistema de archivos de red con los permisos de acceso correctos.

   Una configuración típica es configurar un único directorio compartido que contenga todos los directorios de datos y registros para todos los gestores de colas que utilizan el disco compartido; consulte Compartir qmgrs con nombre y directorios de registros en Configuraciones de directorios de ejemplo en sistemas AIX and Linux.

   Por ejemplo, cree un directorio raíz en la unidad compartida denominado MQHA que tenga subdirectorios `data` y `logs`. Cada gestor de colas crea sus propios directorios de datos y registro en `data` y `logs`. Cree `/MQHA` en la unidad compartida. `/MQHA` es propiedad del usuario y del grupo `mqm` y tiene los permisos de acceso `rwx`.
4. Copie los datos del gestor de colas y los registros a la unidad compartida.

   Siga el procedimiento para realizar una copia de seguridad del gestor de colas descrito en Copia de seguridad de los datos del gestor de colas.

   Nota: A diferencia de Windows, el programa de utilidad hamvmqm no se puede utilizar en AIX and Linux.
5. Actualice la información de configuración del gestor de colas almacenada en el servidor del gestor de colas actual realizando los pasos siguientes:
   a) Modifique la stanza `Log:` en el archivo `qm.ini` del gestor de colas, que está en *share*:

   ```
   LogPath= share/logs/QMgrName
   ```

   b) Modifique la stanza `QueueManager:` en el archivo IBM MQ`mqs.ini`, que normalmente se encuentra en el directorio `/var/mqm` en AIX and Linux:

   ```
   DataPath= share/data/QMgrName
   ```

   Donde *NombreGestorColas* es el nombre de `Directory` en la stanza `QueueManager:` del archivo `mqs.ini` y *share* es la compartición a la que se mueven los datos y registros cronológicos.
6. Añada la información de configuración del gestor de colas al nuevo servidor del gestor de colas.
   a) Ejecute el mandato **dspmqinf** para mostrar la información del gestor de colas.

   Ejecute el mandato en el servidor que ha ejecutado el gestor de colas.

   ```
   dspmqinf -o command QMgrName
   ```

   La salida del mandato tiene el formato listo para crear una configuración del gestor de colas.

   ```
   addmqinf -s QueueManager -v Name= QMgrName -v Directory= QMgrName -v
   Prefix=d:\var\mqm Datapath= \share\data\QMgrName
   ```

   b) Cree una configuración del gestor de colas en el otro servidor.
   Ejecute el mandato **addmqinf** copiado de la salida anterior.
7. Añada la dirección de red del nuevo servidor al nombre de conexión en las definiciones de cliente y de canal.
   a) Busque todos los valores de cliente, emisor y peticionario TCPIP que hacen referencia al servidor.

   Los valores de cliente pueden estar en tablas de definiciones de cliente (CCDT), en variables de entorno, en archivos de propiedades Java o en el código de cliente. Los canales de clúster descubren automáticamente el nombre de conexión de un gestor de colas de su canal receptor de clúster. Mientras el nombre del canal receptor de clúster esté en blanco o se omita, TCPIP descubre la dirección IP del servidor que aloja el gestor de colas.

   b) Modifique el nombre de conexión para que cada una de estas conexiones incluya las direcciones TCPIP de ambos servidores que alojan el gestor de colas de varias instancias.
   Por ejemplo, cambie el nombre de conexión siguiente:

   ```
   echo DISPLAY CHANNEL(ENGLAND) CONNAME | runmqsc QM1
   ```

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
1: DISPLAY CHANNEL(ENGLAND) CONNAME
AMQ8414: Display Channel details.
CHANNEL(ENGLAND) CHLTYPE(SDR)
CONNAME(LONDON)
```

por:

```
echo ALTER CHANNEL(ENGLAND) CHLTYPE(SDR) CONNAME('LONDON, BRISTOL') | runmqsc QM1
```

8. Actualice los procedimientos de supervisión y gestión para que detecten el reinicio del gestor de colas.
9. Actualice las aplicaciones cliente para que se puedan reconectar automáticamente, si procede.
10. Actualice el procedimiento de inicio para que las aplicaciones de IBM MQ se inicien como servicios del gestor de colas.
11. Inicie cada instancia del gestor de colas, permitiéndoles que estén altamente disponibles.

    La primera instancia del gestor de colas que se inicia se convierte en la instancia activa. Emita el mandato dos veces, una vez en cada servidor.

    ```
    strmqm -x QMgrName
    ```

Qué hacer a continuación

Para obtener la más alta disponibilidad de gestores de colas de varias instancias, debe diseñar aplicaciones cliente que sean reconectables y aplicaciones de servidor que sean reiniciables; consulte Recuperación de aplicaciones.

Conceptos relacionados

Recuperación de la aplicación
Reconexión de cliente automática
Reconexión de canal y cliente
Gestores de colas multiinstancia
Archivos de configuración de gestores de colas, qm.ini
Sistema de archivos compartido

Tareas relacionadas

Hacer copia de seguridad de los datos de gestor de colas
Cambio de la información de configuración de IBM MQ en Multiplatforms
Creación de un gestor de colas multiinstancia en Linux
Mover un gestor de colas al almacenamiento de MSCS
Verificación del bloqueo del sistema de archivos compartidos

Referencia relacionada

amqmfsck (comprobación del sistema de archivos)
El archivo de configuración de IBM MQ MQ, mqs.ini

Información relacionada

Prueba de la compatibilidad de un sistema archivos compartidos con gestores de colas de varias instancias de IBM MQ
Declaración de prueba en sistemas de archivos del gestor de colas multiinstancia de IBM MQ

Windows *Conversión de una única instancia en un gestor de colas de varias instancias en Windows*

Para convertir un gestor de colas de una sola instancia, en un gestor de colas de varias instancias, en plataformas Windows , debe mover los datos del gestor de colas a un directorio compartido y volver a configurar el gestor de colas en otros dos servidores.

Antes de empezar

Debe comprobar los requisitos previos para ejecutar un gestor de colas de varias instancias como parte de esta tarea. Para obtener una lista de entornos probados, consulte Declaración de prueba para sistemas de archivos de gestor de colas de varias instancias de IBM MQ. Otros entornos pueden funcionar; se proporciona una herramienta de prueba con IBM MQ para ayudarle a calificar otros entornos.

Debe tener tres servidores que ejecuten un gestor de colas multiinstancia. Un servidor tiene un sistema de archivos compartido para almacenar los datos y registros del gestor de colas. Los demás servidores ejecutan instancias activas y en espera del gestor de colas.

Acerca de esta tarea

Tiene un solo gestor de colas de instancias que desea convertir a un gestor de colas de varias instancias. La conversión del gestor de colas es en sí sencilla, pero debe hacer que otras tareas creen un entorno de producción totalmente automatizado.

Debe comprobar los requisitos previos para un gestor de colas de varias instancias, configurar el entorno y comprobarlo. Debe configurar un sistema da supervisión y de gestión para detectar si el gestor de colas de varias instancias ha fallado y se ha reiniciado automáticamente. A continuación, puede averiguar la causa del reinicio, repararlo y reiniciar el sistema en espera. También debe modificar aplicaciones o el modo en que las aplicaciones están conectadas al gestor de colas, de forma que puedan reanudar el proceso después de un reinicio del gestor de colas.

Procedimiento

1. Compruebe el sistema operativo en el que va a ejecutar el gestor de colas y el sistema de archivos en el que se almacenan los datos y los registros del gestor de colas. Compruebe si pueden ejecutar un gestor de colas de varias instancias.
   a) Consulte Declaración de prueba de sistemas de archivos del gestor de colas multiinstancia de IBM MQ. Consulte si la combinación de sistema operativo y sistema de archivos está probada y es capaz de ejecutar un gestor de colas de varias instancias.

      Un sistema de archivos debe proporcionar un bloqueo basado en arrendamiento para ejecutar gestores de colas de varias instancias. El bloqueo basado en arrendamiento es una característica reciente de algunos sistemas de archivos compartidos y en algunos casos se requieren arreglos. La declaración de soporte le proporciona la información básica.

   b) Ejecute la aplicación de comprobación de integridad que se proporciona en la nota técnica, Prueba de un sistema de archivos compartido para la compatibilidad con gestores de colas de varias instancias de IBM MQ.

      La aplicación de comprobación prueba si el gestor de colas se reinicia correctamente.
2. Configure que un usuario y un grupo puedan acceder a una unidad compartida en el sistema de archivos de red de cada servidor que esté ejecutando una instancia del gestor de colas.

   En Windows, los ID de seguridad (SID) del grupo `mqm` pueden ser diferentes; consulte Windows dominios y gestores de colas de varias instancias de .
3. Configure un directorio para la unidad compartida en el sistema de archivos de red con los permisos de acceso correctos.

   Una configuración típica es configurar un único directorio compartido que contenga todos los directorios de datos y registros para todos los gestores de colas que utilizan el disco compartido; consulte Compartir qmgrs con nombre y directorios de registros .

   Por ejemplo, cree un directorio raíz en la unidad compartida denominado `MQHA` que tenga subdirectorios `data` y `logs`. Cada gestor de colas crea sus propios directorios de datos y registro en `data` y `logs`. Cree *drive* `\MQHA` en la unidad compartida. El propietario es miembro de `mqm`. `mqm` debe tener autorización de control completo. Cree una compartición para *drive*`\MQHA`.
4. Copie los datos del gestor de colas y los registros a la unidad compartida.

   En Windows, puede ejecutar el mandato **hamvmqm** para mover los datos del gestor de colas a la unidad compartida.

5. Añada la información de configuración del gestor de colas al nuevo servidor del gestor de colas.
   a) Ejecute el mandato **dspmqinf** para mostrar la información del gestor de colas
      Ejecute el mandato en el servidor que ha ejecutado el gestor de colas.

      ```
      dspmqinf -o command QMgrName
      ```

      La salida del mandato tiene el formato listo para crear una configuración del gestor de colas.

      ```
      addmqinf -s QueueManager -v Name= QMgrName -v Directory= QMgrName -v
      Prefix=d:\var\mqm Datapath= \share\data\QMgrName
      ```

   b) Cree una configuración del gestor de colas en el otro servidor.
      Ejecute el mandato **addmqinf** copiado de la salida anterior.
6. Añada la dirección de red del nuevo servidor al nombre de conexión en las definiciones de cliente y de canal.
   a) Busque todos los valores de cliente, emisor y peticionario TCPIP que hacen referencia al servidor.
      - Los valores de cliente pueden estar en tablas de definiciones de cliente (CCDT), en variables de entorno, en archivos de propiedades Java o en el código de cliente.
      - Los canales de clúster descubren automáticamente el nombre de conexión de un gestor de colas de su canal receptor de clúster. Mientras el nombre del canal receptor de clúster esté en blanco o se omita, TCPIP descubre la dirección IP del servidor que aloja el gestor de colas.

   b) Modifique el nombre de conexión para que cada una de estas conexiones incluya las direcciones TCPIP de ambos servidores que alojan el gestor de colas de varias instancias.
      Por ejemplo, cambie el nombre de conexión siguiente:

      ```
      echo DISPLAY CHANNEL(ENGLAND) CONNAME | runmqsc QM1
      ```

      ```
      5724-H72 (C) Copyright IBM Corp. 1994, 2024.  ALL RIGHTS RESERVED.
      Starting MQSC for queue manager QM1.
      1: DISPLAY CHANNEL(ENGLAND) CONNAME
      AMQ8414: Display Channel details.
      CHANNEL(ENGLAND) CHLTYPE(SDR)
      CONNAME(LONDON)
      ```

      por:

      ```
      echo ALTER CHANNEL(ENGLAND) CHLTYPE(SDR) CONNAME('LONDON, BRISTOL') | runmqsc QM1
      ```
7. Actualice los procedimientos de supervisión y gestión para que detecten el reinicio del gestor de colas.
8. Actualice las aplicaciones cliente para que se puedan reconectar automáticamente, si procede.
9. Actualice el procedimiento de inicio para que las aplicaciones de IBM MQ se inicien como servicios del gestor de colas.
10. Inicie cada instancia del gestor de colas, permitiéndoles que estén altamente disponibles.
    La primera instancia del gestor de colas que se inicia se convierte en la instancia activa. Emita el mandato dos veces, una vez en cada servidor.

    ```
    strmqm -x QMgrName
    ```

Qué hacer a continuación

Para obtener la más alta disponibilidad de gestores de colas de varias instancias, debe diseñar aplicaciones cliente que sean reconectables y aplicaciones de servidor que sean reiniciables; consulte Recuperación de aplicaciones.

Conceptos relacionados

Recuperación de la aplicación
Reconexión de cliente automática
Reconexión de canal y cliente
Gestores de colas multiinstancia
Archivos de configuración de gestores de colas, qm.ini
Sistema de archivos compartido
Dominios de Windows y gestores de colas multiinstancia

Tareas relacionadas

Hacer copia de seguridad de los datos de gestor de colas
Cambio de la información de configuración de IBM MQ en Multiplatforms
Mover un gestor de colas al almacenamiento de MSCS
Verificación del bloqueo del sistema de archivos compartidos
Trabajar con servicios

Referencia relacionada

amqmfsck (comprobación del sistema de archivos)

Información relacionada

Prueba de la compatibilidad de un sistema archivos compartidos con gestores de colas de varias instancias de IBM MQ
Declaración de prueba en sistemas de archivos del gestor de colas multiinstancia de IBM MQ

Multi *Suprimir un gestor de colas multiinstancia*

En Multiplatforms, para suprimir completamente un gestor de colas multiinstancia, necesita utilizar el mandato **dltmqm** para suprimir el gestor de colas, y a continuación eliminar las instancias de los servidores mediante los mandatos **rmvmqinf** o **dltmqm**.

Ejecute el mandato **dltmqm** para suprimir un gestor de colas que tenga instancias definidas en otros servidores, en cualquier servidor en el que se haya definido ese gestor de colas. No es necesario ejecutar el mandato **dltmqm** en el mismo servidor en el que se haya creado. Ejecute entonces uno de los mandatos **rmvmqinf** o **dltmqm** en el resto de servidores que tengan alguna definición del gestor de colas.

Sólo puede detener el gestor de colas cuando está detenido. En el momento en el que se suprime no hay ninguna instancia ejecutándose, por lo que el gestor de colas, estrictamente hablando, no es un gestor de colas de una sola instancia ni un gestor de colas multiinstancia, sino simplemente un gestor de colas que tiene sus datos y registros en una unidad compartida remota. Cuando se suprime un gestor de colas, se suprimen los datos y registros y su stanza se elimina del archivo `mqs.ini` en el servidor en el que se emita el mandato **dltmqm**. Necesita tener acceso a la unidad compartida de red que contiene los registros y datos del gestor de colas cuando suprima el gestor de colas.

En otros servidores en los que haya creado anteriormente instancias del gestor de colas, también hay entradas en los archivos `mqs.ini` en dichos servidores. Tendrá que eliminar la stanza del gestor de colas en cada servidor ejecutando el mandato **rmvmqinf** *Nombre de la stanza del gestor de colas*.

Linux AIX En sistemas AIX and Linux, si ha colocado un archivo `mqs.ini` común en el almacenamiento de red y le ha hecho referencia desde todos los servidores, estableciendo la variable de entorno `AMQ_MQS_INI_LOCATION` en cada servidor, tiene que suprimir el gestor de colas sólo uno de sus servidores, ya que sólo hay un archivo `mqs.ini` que actualizar.

Ejemplo

Primer servidor

```
dltmqm QM1
```

Otros servidores en los que se esté definida la instancia

```
rmvmqinf QM1 ,o
```

```
dltmqm QM1
```

Linux AIX *Reversión a un gestor de colas de una sola instancia en AIX and Linux*

Revierta un gestor de colas de varias instancias a un gestor de colas de una sola instancia, en AIX and Linux, deteniendo la instancia en espera. A continuación, reinicie la instancia activa y no establezca el distintivo que permite instancias en espera.

Antes de empezar

Tiene al menos tres servidores configurados para ejecutar un gestor de colas como un gestor de colas multiinstancia. El gestor de colas se está ejecutando como un gestor de colas multiinstancia, con una instancia en espera activa.

Acerca de esta tarea

La tarea implica la desactivación de activos en reserva, de modo que sólo el gestor de colas de varias instancias en permanece activo. Para evitar que una instancia en espera se inicie en el futuro, debe detener la instancia activa y reiniciarla. Cuando la reinicie, se inicia como un gestor de colas de una sola instancia que impide que se inicien las instancias en reserva. La instancia en espera está detenida como un paso independiente, para darle la opción de reiniciar la instancia activa en una fecha posterior. Puede detener ambas instancias ejecutando el mandato `endmqm` *QMgrName* estándar en el servidor que ejecuta el gestor de colas activo.

Procedimiento

1. Detenga la instancia del gestor de colas en espera.
   En el servidor que ejecuta la instancia en espera:
   ```
   endmqm -w QMgrName
   ```
2. Detenga la instancia del gestor de colas activa.
   En el servidor que ejecuta la instancia activa:
   ```
   endmqm -w (QMgrName)
   ```
3. Reinicie el gestor de colas, impidiendo sistemas en espera.
   En el servidor que se va a ejecutar el gestor de colas:
   ```
   strmqm QMgrName
   ```

Qué hacer a continuación

Puede que desee ejecutar el gestor de colas como una única instancia en el mismo servidor que los datos del gestor de colas.

Cuando el gestor de colas se detiene, mueva los datos del gestor de colas al servidor que está ejecutando el gestor de colas. O bien instale IBM MQ y, a continuación, mueva la definición de configuración del gestor de colas al servidor con los datos del gestor de colas. Ambas tareas son variaciones de los pasos en “Conversión de una única instancia en un gestor de colas de varias instancias en AIX and Linux” en la página 578 para crear un gestor de colas de varias instancias.

Windows *Reversión a un gestor de colas de una sola instancia en Windows*

Revierta un gestor de colas de varias instancias a un gestor de colas de una sola instancia, en las plataformas Windows, deteniendo la instancia en espera. A continuación, reinicie la instancia activa y no establezca el distintivo que permite instancias en espera.

Antes de empezar

Tiene al menos tres servidores configurados para ejecutar un gestor de colas como un gestor de colas multiinstancia. El gestor de colas se está ejecutando como un gestor de colas multiinstancia, con una instancia en espera activa.

Acerca de esta tarea

La tarea implica la desactivación de activos en reserva, de modo que sólo el gestor de colas de varias instancias en permanece activo. Para evitar que una instancia en espera se inicie en el futuro, debe detener la instancia activa y reiniciarla. Cuando la reinicie, se inicia como un gestor de colas de una sola instancia que impide que se inicien las instancias en reserva. La instancia en espera está detenida como un paso independiente, para darle la opción de reiniciar la instancia activa en una fecha posterior. Puede detener ambas instancias ejecutando el mandato endmqm *QMgrName* estándar en el servidor que ejecuta el gestor de colas activo.

Procedimiento

1. Detenga la instancia del gestor de colas en espera.
   En el servidor que ejecuta la instancia en espera:
   ```
   endmqm -w QMgrName
   ```
2. Detenga la instancia del gestor de colas activa.
   En el servidor que ejecuta la instancia activa:
   ```
   endmqm -w (QMgrName)
   ```
3. Reinicie el gestor de colas, impidiendo sistemas en espera.
   En el servidor que se va a ejecutar el gestor de colas:
   ```
   strmqm QMgrName
   ```

Qué hacer a continuación

Puede que desee ejecutar el gestor de colas como una única instancia en el mismo servidor que los datos del gestor de colas.

Cuando el gestor de colas se detiene, mueva los datos del gestor de colas al servidor que está ejecutando el gestor de colas. O bien instale IBM MQ y, a continuación, mueva la definición de configuración del gestor de colas al servidor con los datos del gestor de colas. Ambas tareas son variaciones de pasos en Conversión de una sola instancia en un gestor de colas de varias instancias en Windows.

Multi *Inicio y detención de un gestor de colas multiinstancia*

Inicio y detención de un gestor de colas configurado en Multiplatforms como una sola instancia o como un gestor de colas multiinstancia.

Cuando haya definido un gestor de colas multiinstancia en un par de servidores, puede ejecutar el gestor de colas en cualquiera de los servidores, ya sea como un gestor de colas de una sola instancia o como un gestor de colas multiinstancia.

Para ejecutar un gestor de colas multiinstancia, inicie el gestor de colas en uno de los servidores mediante el mandato **strmqm** -x *QM1*; la opción -x permite la sustitución por anomalía de la instancia. Se convierte en la *instancia activa*. Inicie la instancia de reserva en el servidor mediante el mismo mandato **strmqm** -x *QM1*; la opción -x permite que la instancia se inicie como instancia de reserva.

El gestor de colas se está ejecutando ahora con una instancia activa que procesa todas las solicitudes y una instancia de reserva que está preparada para sustituir a la instancia activa en caso de que falle. A la instancia activa se le otorga acceso exclusivo a los registros y datos del gestor de colas. La instancia de

reserva espera a que se le otorgue acceso exclusivo a los registros y datos del gestor de colas. Cuando se le otorga acceso exclusivo a la instancia de reserva, pasa a ser la instancia activa.

También puede pasar el control a la instancia de reserva manualmente emitiendo el mandato **endmqm** -s en la instancia activa. El mandato **endmqm** -s termina con la instancia activa sin concluir la de reserva. Se libera el bloqueo de acceso exclusivo en los registros y datos del gestor de colas y la instancia de reserva toma el control.

También puede iniciar y detener un gestor de colas configurado con varias instancias en diferentes servidores como un gestor de colas de una sola instancia. Si inicia el gestor de colas mediante la opción -x en el mandato **strmqm**, las instancias del gestor de colas configuradas en otras maquinas no se inician como instancias de reserva. Si intenta iniciar otra instancia, recibe la respuesta de que no se permite ejecutar la instancia del gestor de colas como una instancia de reserva.

Si detiene la instancia activa de un gestor de colas multiinstancia mediante el mandato **endmqm** sin la opción -s, se detienen ambas instancias, la activa y la de reserva. Si detiene la instancia de reserva mediante el mandato **endmqm** con la opción -x, deja de ser una instancia de reserva y la instancia activa sigue ejecutándose. No puede emitir **endmqm** sin la opción -x en la reserva.

Sólo se pueden ejecutar al mismo tiempo dos instancias del gestor de colas; una es la instancia activa y la otra es la instancia de reserva. Si inicia dos instancias a la vez, IBM MQ no tiene control sobre qué instancia pasa a ser la instancia activa y lo determina el sistema de archivos de red. La primera instancia que obtenga acceso exclusivo a los datos del gestor de colas pasa a ser la instancia activa.

Nota: Antes de reiniciar un gestor de colas que haya fallado, debe desconectar las aplicaciones de dicha instancia del gestor de colas. Si no lo hace, es posible que el gestor de colas no se reinicie correctamente.

Multi *Sistema de archivos compartido*

En Mulitplatforms, un gestor de colas multiinstancia utiliza un sistema de archivos en red para gestionar las instancias de gestor de colas.

Un gestor de colas multiinstancia automatiza la sustitución por anomalía utilizando una combinación de bloqueos de sistema de archivos y datos y registros de gestor de colas compartido. Sólo una instancia de un gestor de colas puede tener acceso exclusivo a los datos y registros de gestor de colas compartidos. Al obtener acceso, se convierte en la instancia activa. La instancia que no haya podido conseguir el acceso exclusivo espera como instancia de reserva hasta que los datos y registros del gestor de colas queden disponibles.

El sistema de archivos en red debe liberar los bloqueos que mantiene para la instancia activa de gestor de colas. Si la instancia activa falla, el sistema de archivos en red libera los bloqueos que mantiene para la instancia activa. En cuanto se libera el bloqueo exclusivo, un gestor de colas de reserva que espera el bloqueo intentará adquirirlo. Si lo consigue, pasa a ser la instancia activa y tiene acceso exclusivo a los datos y registros del gestor de colas en el sistema de archivos compartidos. Continúa el inicio.

El tema relacionado, Planificación del soporte de sistema de archivos describe cómo configurar y comprobar que el sistema de archivos da soporte a los gestores de colas multiinstancia.

Un gestor de colas multiinstancia no le protege contra un error en el sistema de archivos. Existen varias formas de proteger los datos.

- Invertir en un almacenamiento fiable, como matrices redundantes de discos (RAID), e incluirlas en un sistema de archivos en red que tenga resiliencia de red.
- Hacer copia de seguridad de los registros lineales de IBM MQ en un soporte alternativo, y si el soporte de registro primario falla, llevar a cabo la recuperación utilizando los registros del soporte alternativo. Puede utilizar un gestor de colas de copia de seguridad para administrar este proceso.

Multi *Varias instancias de gestores de colas*

Un gestor de colas multiinstancia es resiliente porque utiliza una instancia de gestor de colas de reserva para restaurar la disponibilidad del gestor de colas después de una anomalía.

La duplicación de instancias de gestores de colas es una forma muy efectiva de mejorar las disponibilidad de procesos de gestores de colas. La utilización de un modelo de disponibilidad sencillo, simplemente

como ilustración: si la fiabilidad de una instancia de un gestor de colas es del 99% (en un año, el periodo de inactividad acumulado es de 3,65 días), por lo que añadir otra instancia del gestor de colas aumenta la disponibilidad a un 99.99% (en un año, el periodo de inactividad acumulado es de alrededor de una hora).

Éste es un modelo sencillo que muestra una práctica estimación numérica de la disponibilidad. Para crear una disponibilidad que sea realista, necesita reunir estadísticas del tiempo medio entre anomalías (MTBF) y del tiempo medio en solucionarlas (MTTR), y la distribución de la probabilidad del tiempo entre el tiempo de las anomalías y el tiempo de solucionarlas.

El término gestor de colas multiinstancia hace referencia a la combinación de instancias activas y de reserva del gestor de colas que comparten los registros y los datos del gestor de colas. Los gestores de colas multiinstancia le protegen contra el fallo de los procesos de gestor de colas gracias a tener una instancia del gestor de colas activa en un servidor, y otra instancia del gestor de colas de reserva en otro servidor, preparadas para tomar el control automáticamente si falla la instancia activa.

Multi *Sustitución por anomalía o conmutación*

Una instancia de gestor de colas de reserva sustituye a la instancia activa si se solicita (conmutación) o si la instancia activa falla (sustitución por anomalía).

- La *conmutación* tiene lugar cuando se inicia una instancia en espera en respuesta al mandato **endmqm** `-s` que se emite para la instancia de gestor de colas activa. Se pueden especificar los parámetros **endmqm** `-c`, `-i` o `-p` para controlar la rapidez con la que se detiene el gestor de colas.

  Nota: Las conmutaciones sólo se dan si ya se ha iniciado la instancia del gestor de colas de reserva. El mandato **endmqm** `-s` libera el bloqueo de gestor de colas activo y permite la conmutación: no inicia una instancia de gestor de colas en espera.
- Una *sustitución por anomalía* tiene lugar cuando se libera el bloqueo que mantiene la instancia activa en los datos del gestor de colas debido a que la instancia se detiene de forma inesperada (es decir, sin emitir el mandato **endmqm**).

Cuando la instancia de reserva sustituye a la instancia activa, graba un mensaje en el registro de errores del gestor de colas.

Los clientes reconectables se vuelven a conectar automáticamente cuando el gestor de colas falla o se ha conmutado. No es necesario incluir el indicador `-r` en el mandato **endmqm** para solicitar la reconexión de cliente. La reconexión automática de cliente no está soportada en IBM MQ classes for Java.

Si encuentra que no puede reiniciar una instancia anómala, aunque se haya producido una anomalía y la instancia en espera pase a estar activa, compruebe si las aplicaciones conectadas localmente a la instancia anómala se han desconectado de la instancia anómala.

Las aplicaciones conectadas localmente deben finalizar o desconectarse de una instancia de gestor de colas anómala para que la instancia anómala se reinicie. Las aplicaciones conectadas localmente que utilizan enlaces compartidos (que es el valor predeterminado) que conservan una conexión con una instancia anómala actúan para impedir que la instancia se reinicie.

Si no es posible finalizar las aplicaciones conectadas localmente o garantizar que se desconecten cuando la instancia del gestor de colas local falla, considere la posibilidad de utilizar enlaces aislados. Las aplicaciones conectadas localmente que utilizan enlaces aislados no impiden que se inicie la instancia del gestor de colas local, aunque no se desconecten.

Multi *Reconexión de canal y cliente*

La reconexión de canal y cliente es una parte esencial de la restauración del proceso de mensajes después de que se haya activado una instancia del gestor de colas de reserva.

Las instancias del gestor de colas multiinstancia se instalan en servidores con direcciones de red diferentes. Debe configurar los canales y clientes de IBM MQ con información de conexión para todas las instancias del gestor de colas. Cuando se activa una instancia en espera, los clientes y canales se reconectan automáticamente a la instancia del gestor de colas que acaba de activarse en la dirección de red nueva. La reconexión automática de cliente no está soportada en IBM MQ classes for Java.

El diseño es diferente a los entornos de alta disponibilidad tales como el trabajo HA-CMP. HA-CMP proporciona una dirección IP virtual para el clúster y transfiere la dirección al servidor activo. La reconexión de IBM MQ no cambia ni redirecciona las direcciones IP. Funciona reconectando mediante las direcciones de red que se han definido en las definiciones de canal y conexiones de cliente. Como administrador, debería definir las direcciones de red en las definiciones de canal y conexiones cliente en todas las instancias de cualquier gestor de colas multiinstancia. La mejor forma de configurar direcciones de red en un gestor de colas multiinstancia depende de la conexión:

Canales de gestores de colas
El atributo de canales CONNAME es una lista de nombres de conexiones separados por comas; por ejemplo, `CONNAME('127.0.0.1(1234), 192.0.2.0(4321)')`. Las conexiones se intentan en el orden en el que están especificadas en la lista de conexiones hasta que una de ellas se establece de forma satisfactoria. Si no hay ninguna que se establezca de forma satisfactoria, el canal intenta la reconexión.

Canales de clúster

Normalmente no se requiere ninguna configuración adicional para que funcionen los gestores de colas multiinstancia en un clúster.

Si un gestor de colas se conecta a un gestor de colas de repositorio, el repositorio descubre la dirección de red del gestor de colas. Se hace referencia a la opción CONNAME del canal CLUSRCVR en el gestor de colas. En TCPIP, el gestor de colas establece automáticamente la opción CONNAME si el usuario la omite, o la configura en blanco. Cuando una instancia de reserva pasa a ser la activa, su dirección de IP sustituye a la de la instancia activa anterior como la opción CONNAME.

Si es necesario, puede configurar manualmente la opción CONNAME con la lista de direcciones de red de las instancias del gestor de colas.

Conexiones del cliente
Las conexiones de cliente pueden utilizar listas de conexiones o grupos de gestores de colas para seleccionar conexiones alternativas.

Cuando ocurre alguna anomalía, la reconexión lleva algún tiempo. El gestor de colas de reserva tiene que terminar de iniciarse. Los clientes que se hayan conectado al gestor de colas que ha fallado tienen que detectar la anomalía de la conexión, e iniciar una nueva conexión con el cliente. Si una nueva conexión con el cliente selecciona el gestor de colas de reserva que se ha activado recientemente, entonces el cliente vuelve a conectarse al mismo gestor de colas.

Si el cliente está en medio de una llamada MQI durante una reconexión, debe tolerar una espera larga antes de que la llamada se complete.

Si la anomalía tiene lugar durante una transferencia por lotes en un canal de mensajes, se restituye el lote y se vuelve a iniciar.

El cambio es más rápido que la sustitución por anomalía y sólo dura el tiempo en el que se detiene una instancia del gestor de colas y se inicia otra. Para un gestor de colas que tenga que reproducir sólo un número pequeño de registros, el tiempo mínimo que puede tardar en conmutarse es de unos segundos. Para estimar cuánto tiempo tarda la sustitución por anomalía, debe añadir el tiempo que se tarda en detectar la anomalía. En el mejor de los casos esto puede durar unos diez segundos y puede llegar a varios minutos, dependiendo de la red y del sistema de archivos.

Multi *Recuperación de la aplicación*

La recuperación de la aplicación es la continuación automática del proceso de la aplicación tras la sustitución por anomalía. La recuperación de la aplicación tras una sustitución por anomalía necesita un diseño minucioso. Algunas aplicaciones necesitan saber que la sustitución por anomalía ha tenido lugar.

La recuperación de la aplicación tiene como objetivo que la aplicación continúe procesando tras un breve retraso. Antes de continuar con un nuevo proceso, la aplicación deberá restituir y reenviar la unidad de trabajo que estaba procesando en el momento de la anomalía.

Un problema para la recuperación de aplicación es perder el contexto que se comparte entre el IBM MQ MQI client y el gestor de colas y que se almacena en el gestor de colas. El IBM MQ MQI client

restaura la mayor parte del contexto, pero hay algunas partes del contexto que no se pueden restaurar de forma fiable. Las secciones siguientes describen algunas propiedades de la recuperación de la aplicación y la manera en la que afectan a la recuperación de aplicaciones conectadas a un gestor de colas multiinstancia.

Mensajería transaccional

Desde una perspectiva de entrega de mensajes, la sustitución por anomalía no cambia las propiedades persistentes de la mensajería de IBM MQ. Si los mensajes son persistentes y se gestionan correctamente dentro de unidades de trabajo, no se pierden durante una sustitución por anomalía.

Desde una perspectiva de proceso de transacción, las transacciones se restituyen o confirman tras la sustitución por anomalía.

Las transacciones no confirmadas se restituyen. Después de la migración tras error, una aplicación reconectable recibe el código de razón `MQRC_BACKED_OUT` para indicar que la transacción ha fallado. Se necesita entonces volver a iniciar la transacción de nuevo.

Las transacciones confirmadas son aquellas que han alcanzado la segunda fase de una confirmación de dos fases o transacciones de una sola fase (sólo mensajes) que empiezan con `MQCMIT`.

Si el gestor de las colas es el coordinador de transacciones y `MQCMIT` ha empezado la segunda fase de su confirmación de dos fases antes de la anomalía, la transacción se completará correctamente. La fase de terminación está bajo el control del gestor de colas y continúa cuando el gestor de colas se vuelve a ejecutar. En una aplicación reconectable, la llamada `MQCMIT` se completa de forma normal.

En una confirmación de solo una fase, que afecta sólo a mensajes, una transacción que ha iniciado el proceso de confirmación se completa normalmente bajo el control del gestor de colas en cuanto se ejecute de nuevo. En una aplicación reconectable, `MQCMIT` se completa de forma normal.

Los clientes reconectables puede utilizar transacciones de fase única bajo el control del gestor de colas como coordinador de transacciones. El cliente transaccional extendido no admite la reconexión. Si la reconexión se solicita cuando el cliente se conecta, la conexión es correcta, pero no tiene disponibilidad para poder volver a conectarse. La conexión se comporta como si no fuera reconectable.

La aplicación se reinicia o se reanuda

La sustitución por anomalía interrumpe una aplicación. Después de una anomalía, una aplicación se puede reanudar desde el principio o puede retomar el proceso tras la interrupción. Esto último se denomina *reconexión automática de cliente*. La reconexión automática de cliente no está soportada en IBM MQ classes for Java.

Con una aplicación de IBM MQ MQI client, puede establecer una opción de conexión para reconectar automáticamente el cliente. Las opciones son `MQCNO_RECONNECT` o `MQCNO_RECONNECT_Q_MGR`. Si no se establece ninguna opción, el cliente no intentará reconectarse de forma automática y la anomalía del gestor de colas devolverá `MQRC_CONNECTION_BROKEN` al cliente. Puede diseñar el cliente para probar e iniciar una nueva conexión emitiendo una nueva llamada `MQCONN` o `MQCONNX`.

Los programas de servidor tiene que reiniciarse; el gestor de colas no puede volver a conectarlos de forma automática a partir del punto en el que estaban cuando se produjo el error en el gestor de colas o en el servidor. Los programas de servidor de IBM MQ no suelen reiniciarse en la instancia del gestor de colas en espera cuando falla una instancia del gestor de colas multiinstancia.

Puede automatizar un programa de servidor de IBM MQ para que se reinicie en el servidor en espera de dos maneras:

1. Empaquetar la aplicación de servidor como un servicio de gestor de colas. Se reinicia cuando se reinicia el gestor de colas de reserva.
2. Escribir una lógica de sustitución por anomalía propia, desencadenada por ejemplo por el mensaje de registro de sustitución por anomalía grabado por una instancia del gestor de colas de reserva cuando se inicia. La instancia de la aplicación necesita entonces llamar a `MQCONN` o `MQCONNX` tras iniciarse para crear una conexión con el gestor de colas.

Supresión de anomalías

Algunas aplicaciones deben tener conocimiento de la sustitución por anomalía, otras no. Observe los siguientes dos ejemplos.

1. Una aplicación de mensajería que obtiene o recibe mensajes por un canal de mensajería no suele necesitar que el gestor de colas en el otro extremo del canal esté ejecutándose: no es probable que se vea afectado si el gestor de colas en el otro extremo del canal se reinicia en una instancia de reserva.
2. Una aplicación IBM MQ MQI client procesa entrada de mensajes persistentes de una cola y pone respuestas de mensajes persistentes en otra cola como parte de una única unidad de trabajo: si se recibe un código de razón MQRC_BACKED_OUT procedente de MQPUT, MQGET o MQCMIT dentro de un punto de sincronización reiniciando la unidad de trabajo, no se pierde ningún mensaje. Además la aplicación no necesita realizar ningún proceso especial ante la anomalía de conexión.

Suponga sin embargo, en este segundo ejemplo, que la aplicación examina la cola para seleccionar el mensaje que va a procesar utilizando la opción MQGET, MQGMO_MSG_UNDER_CURSOR. La reconexión establece el cursor para examinar y la llamada MQGET no devuelve el mensaje correcto. En este ejemplo, la aplicación debe conocer que se ha producido la anomalía. Adicionalmente, antes de emitir otra MQGET para el mensaje señalado por el cursor, la aplicación tiene que restablecer el cursor para examinar.

La pérdida del cursor para examinar es un ejemplo de cómo el contexto de aplicación cambia después de la reconexión. En "Recuperación de un cliente reconectado automáticamente" en la página 590 pueden encontrase otros casos documentados.

Tiene tres patrones de diseño alternativos para aplicaciones de IBM MQ MQI client después de una sustitución por anomalía. Sólo uno de ellos no necesita detectar la sustitución por anomalía.

Sin reconexión

En este patrón, la aplicación detiene todo el proceso en la conexión actual cuando se interrumpe la conexión. Para que la aplicación continúe el proceso, debe establecer una nueva conexión con el gestor de colas. La aplicación es totalmente responsable de transferir cualquier información de estado que necesite para continuar el proceso en la nueva conexión. Las aplicaciones de cliente existentes que se reconectan con el gestor de colas tras perder la conexión se graban de esta forma.

El cliente recibe un código de razón, como por ejemplo MQRC_CONNECTION_BROKEN o MQRC_Q_MGR_NOT_AVAILABLE desde la siguiente llamada MQI tras perder la conexión. La aplicación debe descartar toda la información de estado de IBM MQ, como manejadores de colas, y emitir una nueva llamada MQCONN o MQCONNX para establecer una nueva conexión y luego reabrir los objetos de IBM MQ que deba procesar.

De forma predeterminada, el comportamiento de MQI es que el manejador de conexión del gestor de colas ya no funcione tras perder la conexión con el gestor de colas. Esto equivale a establecer la opción MQCNO_RECONNECT_DISABLED en MQCONNX, previniendo así la reconexión de la aplicación después de que se produzca la sustitución por anomalía.

Tolerante a anomalías

Grabe la aplicación para que no se vea afectada por la anomalía. En ocasiones, basta con manejar con cuidado los errores para solucionar la anomalía.

Preparado para reconexión

Registre un manejador de sucesos MQCBT_EVENT_HANDLER con el gestor de colas. El manejador de sucesos se manda con MQRC_RECONNECTING cuando se inicia el cliente para volver a conectar con el servidor, y con MQRC_RECONNECTED después de una reconexión satisfactoria. Puede ejecutar entonces una rutina para restablecer un estado previsible para que la aplicación cliente pueda continuar el proceso.

Recuperación de un cliente reconectado automáticamente

La sustitución por anomalía es un suceso inesperado, y para que un cliente reconectado automáticamente funcione tal como se ha diseñado, las consecuencias de la reconexión deben ser predecibles.

El uso de transacciones posibilita más la conversión de una anomalía inesperada en una recuperación previsible y fiable.

En la sección anterior se mostraba un ejemplo, “2” en la página 590, de un IBM MQ MQI client que utiliza una transacción local para coordinar MQGET y MQPUT. El cliente emite una llamada MQCMIT o MQBACK en respuesta a un error MQRC_BACKED_OUT y luego reenvía la transacción restituida. La anomalía del gestor de colas provoca la restitución de la transacción y el comportamiento de la aplicación del cliente garantiza que no se pierdan transacciones ni mensajes.

Tenga en cuenta que para una devolución de llamada, es posible que sea necesario reanudar la aplicación consumidora si el estado del parámetro de devolución de llamada es: MQCS_SUSPENDED_USER_ACTION.

No todos los estados de programas se gestionan como parte de una transacción y, por ello, las consecuencias de la reconexión son más difíciles de comprender. Debe saber cómo la reconexión cambia el estado de un IBM MQ MQI client para diseñar su aplicación cliente para superar una sustitución por anomalía del gestor de colas.

Puede diseñar su aplicación sin ningún código de sustitución por anomalía especial, gestionando los errores de reconexión con la misma lógica que para otros errores. De forma alternativa, puede optar por reconocer que la reconexión requiere un proceso de errores especial y registrar un manejador de sucesos en IBM MQ para que ejecute una rutina para manejar la sustitución por anomalía. La rutina puede manejar por sí misma el proceso de reconexión o bien establecer un distintivo para indicar a la hebra de programa principal que cuando reanude el proceso, debe realizar el proceso de recuperación.

El propio entorno de IBM MQ MQI client tiene conocimiento de la sustitución por anomalía, y restaura todo el contexto posible, tras la reconexión, almacenando parte de la información de estado en el cliente, y emitiendo llamadas MQI adicionales en nombre de la aplicación cliente para restaurar el estado de IBM MQ. Por ejemplo, se restauran los manejadores de objetos que estaban abiertos en el punto de anomalía y se abren colas dinámicas temporales con el mismo nombre. No obstante hay cambios que son inevitables y su configuración tiene que hacer frente a estos cambios. Las diferencias pueden ser de cinco tipos:

1. Las llamadas MQI devuelven errores nuevos o no diagnosticados previamente hasta que el programa de aplicación establece un nuevo estado de contexto consistente.

   Un ejemplo de recibir un error nuevo es el código de retorno MQRC_CONTEXT_NOT_AVAILABLE al intentar pasar contexto después de guardar contexto antes de la reconexión. No se puede restaurar el contexto después de la reconexión porque el contexto de seguridad no ha pasado a un programa cliente no autorizado. Hacerlo permitiría obtener el contexto de seguridad a un programa de aplicación malicioso.

   Normalmente las aplicaciones manejan los errores previsibles y comunes diseñados cuidadosamente y relegan otros errores no comunes a un manejador de errores genérico. El manejador de errores puede desconectarse de IBM MQ y conectarse de nuevo, o incluso detener totalmente el programa. Para mejorar la continuidad es posible que tenga que tratar los errores de forma distinta.
2. Es posible que se pierdan mensajes no persistentes.
3. Las transacciones se retrotraen (que también pueden suspender consumidores asíncronos, consulte el texto anterior).
4. Las llamadas MQGET o MQPUT utilizadas fuera de un punto de sincronización pueden interrumpirse con posible pérdida de algún mensaje.
5. El tiempo provoca errores, por una espera prolongada en una llamada MQI.

A continuación se enumeran algunos detalles sobre la pérdida de contexto.

- Los mensajes no persistentes se descartan, si no se transfieren a una cola con la opción NPMCLASS(HIGH), y la anomalía del gestor de colas no interrumpa la opción de almacenamiento en conclusión.
- Una suscripción no duradera es una pérdida cuando se interrumpe una conexión. Durante la reconexión, se restablece. Considere el uso de una suscripción duradera.

- El intervalo obtener-esperar se vuelve a calcular; si se supera el límite, se devuelve MQRC_NO_MSG_AVAILABLE. Se calcula de forma similar la caducidad de suscripción para dar el tiempo de caducidad global.
- La posición del cursor para examinar en una cola se pierde; normalmente se restablece antes del primer mensaje.
  - Las llamadas MQGET que especifican MQGMO_BROWSE_MSG_UNDER_CURSOR o MQGMO_MSG_UNDER_CURSOR tienen una anomalía con código de razón MQRC_NO_MSG_AVAILABLE.
  - Los mensajes bloqueados para examinar se desbloquean.
  - Examinar los mensajes marcados con ámbito de manejador que no están marcados y pueden examinarse de nuevo.
  - Examinar de forma cooperativa mensajes marcados y no marcados en la mayoría de los casos.
- Se pierde el contexto de seguridad. Intenta utilizar contexto de mensajes guardados, como transferir un mensaje con la anomalía MQPMO_PASS_ALL_CONTEXT con MQRC_CONTEXT_NOT_AVAILABLE.
- Se pierden señales de mensajes. MQGET devuelve el código de razón MQRC_NO_MSG_AVAILABLE al utilizar una señal de mensajes.

  Nota: *MsgId* y *CorrelId*, ya que forman parte del mensaje, se conservan con el mensaje durante la migración tras error y, por lo tanto, MQGET utilizan MsgId o CorrelId funcionan según lo esperado.
- Los mensajes colocados en una cola bajo el punto de sincronización en una transacción sin confirmar ya no están disponibles.
- El proceso de mensajes en un orden lógico, o en un grupo de mensajes, da domo resultado un código de retorno de MQRC_RECONNECT_INCOMPATIBLE tras la reconexión.
- Una llamada MQI puede devolver MQRC_RECONNECT_FAILED en vez del más general MQRC_CONNECTION_BROKEN que los clientes suelen recibir actualmente.
- La reconexión durante una llamada MQPUT fuera del punto de sincronización devuelve MQRC_CALL_INTERRUPTED si el IBM MQ MQI client no sabe si el mensaje se ha entregado de forma satisfactoria al gestor de colas. La reconexión durante MQCMIT se comporta de igual modo.
- Se devuelve MQRC_CALL_INTERRUPTED - tras una reconexión satisfactoria - si el IBM MQ MQI client no ha recibido ninguna respuesta del gestor de colas para indicar el éxito o el fracaso de
  - la entrega de un mensaje persistente utilizando una llamada MQPUT fuera del punto de sincronización.
  - la entrega de un mensaje persistente o un mensaje con persistencia predeterminada utilizando una llamada MQPUT1 fuera del punto de sincronización
  - la confirmación de una transacción utilizando una llamada MQCMIT. La respuesta sólo se devuelve después de efectuar una reconexión satisfactoria.
- Los canales se reinician como nuevas instancias (pueden ser también diferentes canales) y no se retiene ningún estado de salida de canal.
- Las colas dinámicas temporales se almacenan como parte del proceso de recuperación de clientes reconectables que han tenido colas dinámicas temporales abiertas. No se restablecen los mensajes de una cola dinámica temporal, pero las aplicaciones que han tenido la cola abierta, o han recordado el nombre de la cola, pueden continuar con el proceso.

  Existe la posibilidad de que si la cola la utiliza una aplicación que no es la que la creó, no se pueda restablecer suficientemente rápido para que esté presenta en la siguiente referencia. Por ejemplo, si un cliente crea una cola dinámica temporal como una cola de respuesta, y un mensaje de respuesta debe colocarse en la cola mediante un canal, es posible que la cola no se pueda recuperar a tiempo. En ese caso, el canal colocaría el mensaje de respuesta en la cola de mensajes no entregados.

  Si una aplicación cliente reconectable abre una cola dinámica temporal por nombre (debido a que otra aplicación ya la ha creado), cuando se produce la reconexión, el IBM MQ MQI client no puede volver a crear la cola dinámica temporal porque no tiene el modelo desde el que crearla. En la MQI, sólo una aplicación puede abrir la cola dinámica temporal por el modelo. Otras aplicaciones que deseen utilizar

la cola dinámica temporal deben utilizar MQPUT1 o los enlaces del servidor o ser capaces de intentar la reconexión de nuevo si falla.

Sólo se pueden transferir a una cola dinámica temporal los mensajes no persistentes y estos se pierden durante la sustitución por anomalía; esta pérdida se cumple para mensajes que se transfieren a una cola dinámica temporal utilizando MQPUT1 durante la reconexión. Si la anomalía se produce durante MQPUT1, puede que el mensaje no se transfiera, aunque la operación MQPUT1 se realice correctamente. Una solución a este problema es utilizar colas dinámicas permanentes. Cualquier aplicación de enlaces del servidor puede abrir la cola dinámica temporal por el nombre porque no es reconectable.

Multi *Recuperación de datos y alta disponibilidad*

Las soluciones de alta disponibilidad que utilizan gestores de colas multiinstancia necesitan incluir un mecanismo para recuperar datos después de una anomalía de almacenamiento.

Un gestor de colas multiinstancia aumenta la disponibilidad de procesos de gestores de colas, pero no la de otros componentes, como el sistema de archivos que el gestor de colas utiliza para almacenar mensajes y alguna otra información.

Una forma de hacer los datos altamente disponibles es utilizar un almacenamiento de datos resiliente en red. Puede crear una solución propia utilizando un sistema de archivos en red y un almacenamiento de datos resiliente, o puede comprar una solución integrada. Si desea combinar la flexibilidad con la recuperación de desastres, está disponible la duplicación de disco asíncrona, la cual permite la duplicación de disco sobre decenas y cientos de kilómetros.

Puede configurar la forma en que diferentes directorios de IBM MQ se correlacionan con soportes de almacenamiento, para aprovechar lo mejor posible el soporte. Para los gestores de colas *multiinstancia* existe una distinción importante entre dos tipos de directorios y archivos de IBM MQ.

Directorios que deben compartirse entre las instancias de un gestor de colas.
La información que debe compartirse entre distintas instancias de un gestor de colas está en dos directorios: los directorios `qmgrs` y `logs`. Los directorios deben ser un sistema de archivos interconectados compartido. Se recomienda utilizar un soporte de almacenamiento que proporcione una gran disponibilidad continua y un rendimiento excelente porque los datos cambian de forma constante a medida que se crean y suprimen mensajes.

Directorios y archivos que no *tienen* que compartirse entre instancias de un gestor de colas.
Existen otros directorios que no tienen que compartirse entre distintas instancias de un gestor de colas y se restauran rápidamente mediante otros sistemas distintos al sistema de archivos duplicados.

- Archivos ejecutables de IBM MQ y el directorio de herramientas. Sustituir reinstalando o haciendo copia de seguridad y restaurando desde un archivador de archivos de copia de seguridad.
- Información de configuración que se modifica para toda la instalación. La información de configuración es gestionada por IBM MQ, como el archivo `mqs.ini` en sistemas AIX, Linux, and Windows, o parte de su propia gestión de configuración como, por ejemplo, scripts de configuración de **MQSC** . Hacer una copia de seguridad y restablecer utilizando un archivado de archivos.
- Salida para toda la instalación como rastreos, registros de error y archivos FFDC. Los archivos se almacenan en los subdirectorios `errors` y `trace` en el directorio de datos predeterminado. El directorio de datos predeterminado en sistemas AIX and Linux es `/var/mqm`. En Windows, el directorio de datos predeterminado es el directorio de instalación de IBM MQ.

También puede utilizar una gestor de colas de seguridad para realizar regularmente copias de seguridad de soportes de un gestor de colas multiinstancia utilizando el registro lineal. Un gestor de colas de copia de seguridad no ofrece una recuperación tan rápida como un sistema de archivos y no recupera los cambios desde la última copia de seguridad. Es más apropiado utilizar el mecanismo del gestor de colas de copia de seguridad para escenarios que se recuperan de desastres exteriores que recuperar un gestor de colas tras una anomalía de almacenamiento localizada.

Combinación de soluciones de disponibilidad de IBM MQ

Las aplicaciones utilizan otras funciones de IBM MQ para mejorar la disponibilidad. Los gestores de colas multiinstancia complementan otras funciones de alta disponibilidad.

Los clústeres de IBM MQ aumentan la disponibilidad de colas

Se puede aumentar la disponibilidad de colas creando varias definiciones de una cola de clústeres; hasta uno por cada cola en cada gestor del clúster.

Imagine que un miembro del clúster falla y se envía un mensaje nuevo a una cola de clústeres. A menos que el mensaje *tenga* que ir al gestor de colas que ha fallado, el mensaje se envía a otro gestor de colas que se esté ejecutando en el clúster que tenga una definición de la cola.

Aunque los clústeres aumentan bastante la disponibilidad, existen dos escenarios de anomalías relacionados en los que los mensajes se retrasan. La creación de un clúster con gestores de colas multiinstancia reduce la posibilidad de que se retrasen los mensajes.

Mensajes abandonados

Si falla un gestor de colas en el clúster, el resto de mensajes que puedan direccionarse a otros gestores de colas en el clúster no vuelven a direccionarse al gestor de colas que ha fallado. Los mensajes que se han enviado se retienen hasta que el gestor de colas que ha fallado vuelve a reiniciarse.

Afinidades

Afinidad es el término que se utiliza para describir la información compartida entre otros dos cálculos separados. Por ejemplo, se da una afinidad entre una aplicación que envía un mensaje de solicitud a un servidor y la misma aplicación que debería procesar la respuesta. Otros ejemplo sería una secuencia de mensajes, el proceso de cada mensaje depende de los mensajes anteriores.

Si envía mensajes a colas con clústeres, debe considerar las afinidades. ¿Es necesario enviar mensajes sucesivos al mismo gestor de colas o puede ir cada mensaje a cualquier miembro del clúster?

Si necesita enviar mensajes al mismo gestor de colas en el clúster y este falla, los mensajes esperan en la cola de transmisión del emisor hasta que el gestor de colas del clúster que ha fallado se ejecuta de nuevo.

Si el clúster se configura con gestores de colas multiinstancia, el tiempo de retraso en espera de que el gestor de colas anómalo vuelva a iniciarse es de más o menos un minuto hasta que el de espera lo sustituye. Cuando se está ejecutando el de espera, los mensajes retenidos continúan el proceso, se inician los canales al gestor de colas que ha pasado a estar activo recientemente y los mensajes que se encontraban esperando en las colas de transmisión empiezan a fluir.

Una posible forma de configurar un clúster para que reduzca los mensajes que se retrasan cuando falla un gestor de colas es desplegar dos gestores de colas diferentes para cada servidor en el clúster y poner uno activo y otro en espera de los diferentes gestores de colas. Ésta es una configuración activa/en espera y aumenta la disponibilidad del clúster.

Además de reducir la administración y aumentar la escalabilidad, los clústeres continúan proporcionando elementos de disponibilidad adicionales para complementar los gestores de colas multiinstancia. Los clústeres ofrecen protección frente a otro tipo de anomalías que afectan a instancias activas y pasivas de un gestor de colas.

Servicio sin interrupción

Un clúster ofrece un servicio sin interrupción. El clúster envía los nuevos mensajes que recibe a los gestores de colas activos para que los procese. No confíe en un gestor de colas multiinstancia para que proporcione un servicio sin interrupción, ya que el gestor de colas de reserva tarda un tiempo en detectar la anomalía y completar su inicio, los canales tardan un tiempo en reconectarse y los lotes de mensajes que tienen anomalías tardan un tiempo en volverse a enviar.

Interrupción local
Existen limitaciones prácticas en la distancia que puede haber entre los servidores de sistemas de archivos activos y en espera, ya que debe interactuar a una velocidad de milisegundos para ofrecer un rendimiento aceptable.

Los gestores de colas en clúster necesitan velocidades de interacción de muchos segundos y pueden encontrase en cualquier sitio del mundo.

Errores operativos
Al utilizar dos mecanismos diferentes para aumentar la disponibilidad, se reducen las oportunidades de que un error operativo, como puede ser un error humano, termine con todo los esfuerzos realizados para obtener disponibilidad.

Los grupos compartidos de colas aumentan la disponibilidad del proceso de mensajes

z/OS Los grupos de compartición de colas, que sólo se proporcionan en z/OS, permiten a un grupo de gestores de colas compartir el servicio a una cola. Si algún gestor de colas tiene alguna anomalía, los otros gestores de colas continúan procesando todos los mensajes de la cola. Los gestores de colas multiinstancia no están soportados en z/OS y complementan los grupos de compartición de colas sólo como parte de una arquitectura de mensajería más amplía.

Los clientes de IBM MQ aumentan la disponibilidad de las aplicaciones

Los programas de IBM MQ MQI client pueden conectarse a diferentes gestores de colas en un grupo de gestores de colas en función de la disponibilidad de gestores de colas, las ponderaciones de conexiones y las afinidades. Al ejecutar una aplicación en una máquina diferente a la que se está ejecutando el gestor de colas, se puede aumentar la disponibilidad total de una solución en la medida que haya una forma de reconectar la aplicación si la instancia del gestor de colas que está conectada falla.

Los grupos de gestores de colas también se utilizan para aumentar la disponibilidad de clientes separando un cliente del gestor de colas que se ha detenido, y el balance de colas de las conexiones de clientes por todo un grupo de gestores de colas, en vez de la técnica de IP spraying. La aplicación cliente no debe tener afinidades con el gestor de colas que ha fallado, como una dependencia en una cola específica, ya que no podrá continuar el proceso.

La reconexión automática de cliente y los gestores de colas multiinstancia aumentan la disponibilidad de clientes resolviendo algunos problemas de afinidad. La reconexión automática de cliente no está soportada en IBM MQ classes for Java.

Se puede establecer la opción `MQCNO` `MQCNO_RECONNECT_Q_MGR`, para forzar a que un cliente se conecte al mismo gestor de colas:

1. Si el gestor de colas de una única instancia conectado previamente no se está ejecutando, la conexión se intenta de nuevo hasta que el gestor de colas se esté ejecutando de nuevo.
2. Si el gestor de colas se configura como un gestor de colas multiinstancia, el cliente se conecta a cualquier instancia que esté activa.

Con la reconexión automática al mismo gestor de colas, se restablece mucha de la información de estado que el gestor de colas mantenía sobre el cliente, como las colas que tiene abiertas y el tema al que está suscrito. Si el cliente tenía abierto una cola de respuesta dinámica para recibir una respuesta a una solicitud, la conexión a la cola de respuesta se restaura también.

MQ Adv. Linux Alta disponibilidad en RDQM

RDQM (gestor de colas de datos replicados) es una solución de alta disponibilidad que está disponible en las plataformas Red Hat Enterprise Linux for x86-64 .

Una configuración RDQM consta de tres servidores configurados en un grupo de alta disponibilidad (HA), cada uno con una instancia del gestor de colas. Una instancia es el gestor de colas en ejecución, que replica síncronamente sus datos en las otras dos instancias. Si el servidor que está ejecutando este gestor de colas falla, se inicia otra instancia del gestor de colas que tiene datos actuales con los que operar. Las

tres instancias del gestor de colas pueden compartir opcionalmente una dirección IP flotante, por lo que solo es necesario configurar los clientes con una única dirección IP. En un determinado momento solo puede ejecutar una única instancia del gestor de colas, incluso si el grupo HA se particiona por problemas de red. El servidor que ejecuta el gestor de colas se conoce como 'primario', mientras que cada uno de los otros dos servidores se conoce como 'secundario'.

Se utilizan tres nodos para reducir considerablemente la posibilidad de que surja una situación de cerebro dividido. En un sistema de alta disponibilidad de dos nodos la situación de cerebro dividido puede suceder cuando la conectividad entre los dos nodos se interrumpe. Sin conectividad, los dos nodos podrían ejecutar el gestor de colas simultáneamente, lo que acumularía datos distintos. Cuando se restaura la conexión, hay dos versiones distintas de los datos (un 'cerebro dividido') y se requiere la intervención manual para decidir qué conjunto de datos se debe mantener y cuál se debe descartar.

RDQM utiliza un sistema de tres nodos con quórum para evitar la situación de cerebro dividido. Los nodos que se pueden comunicar con al menos uno de los otros nodos forman un quórum. Los gestores de colas solo se pueden ejecutar en un nodo que tenga quórum. El gestor de colas no se puede ejecutar en un nodo que no esté conectado al menos a otro nodo, de manera que nunca se puede ejecutar en dos nodos simultáneamente:

- Si falla un solo nodo, el gestor de colas se puede ejecutar en uno de los otros dos nodos. Si fallan dos nodos, el gestor de colas no se puede ejecutar en el nodo que queda porque ese nodo no tiene quórum (este nodo no puede saber si los otros dos nodos han fallado o si aún se están ejecutando y ha perdido la conectividad).
- Si pierde la conectividad un solo nodo, el gestor de colas no se puede ejecutar en este nodo porque el nodo no tiene quórum. El gestor de colas se puede ejecutar en uno de los dos nodos restantes, que sí tienen quórum. Si pierden la conectividad todos los nodos, el gestor de colas no se puede ejecutar en ninguno de los nodos, ya que ninguno de ellos tiene quórum.

Nota: La IBM MQ Console no soporta los gestores de colas replicados. Puede utilizar IBM MQ Explorer con gestores de colas de datos replicados, pero esto no muestra información específica de las características RDQM.

La configuración del grupo de los tres nodos se gestiona con Pacemaker. La réplica entre los tres nodos la gestiona DRBD. (Consulte https://clusterlabs.org/pacemaker/ para obtener información sobre Pacemaker y https://docs.linbit.com/docs/users-guide-9.0/ para obtener información sobre DRBD.)

Puede hacer una copia de seguridad de los gestores de colas de datos duplicados utilizando el proceso descrito en “Hacer copia de seguridad de los datos de gestor de colas” en la página 707. La detención del gestor de colas y realizar una copia de seguridad del mismo no tiene ningún efecto sobre la supervisión del nodo realizada por la configuración de RDQM.

La figura siguiente muestra un despliegue típico con un RDQM ejecutando en cada uno de los tres nodos del grupo HA.

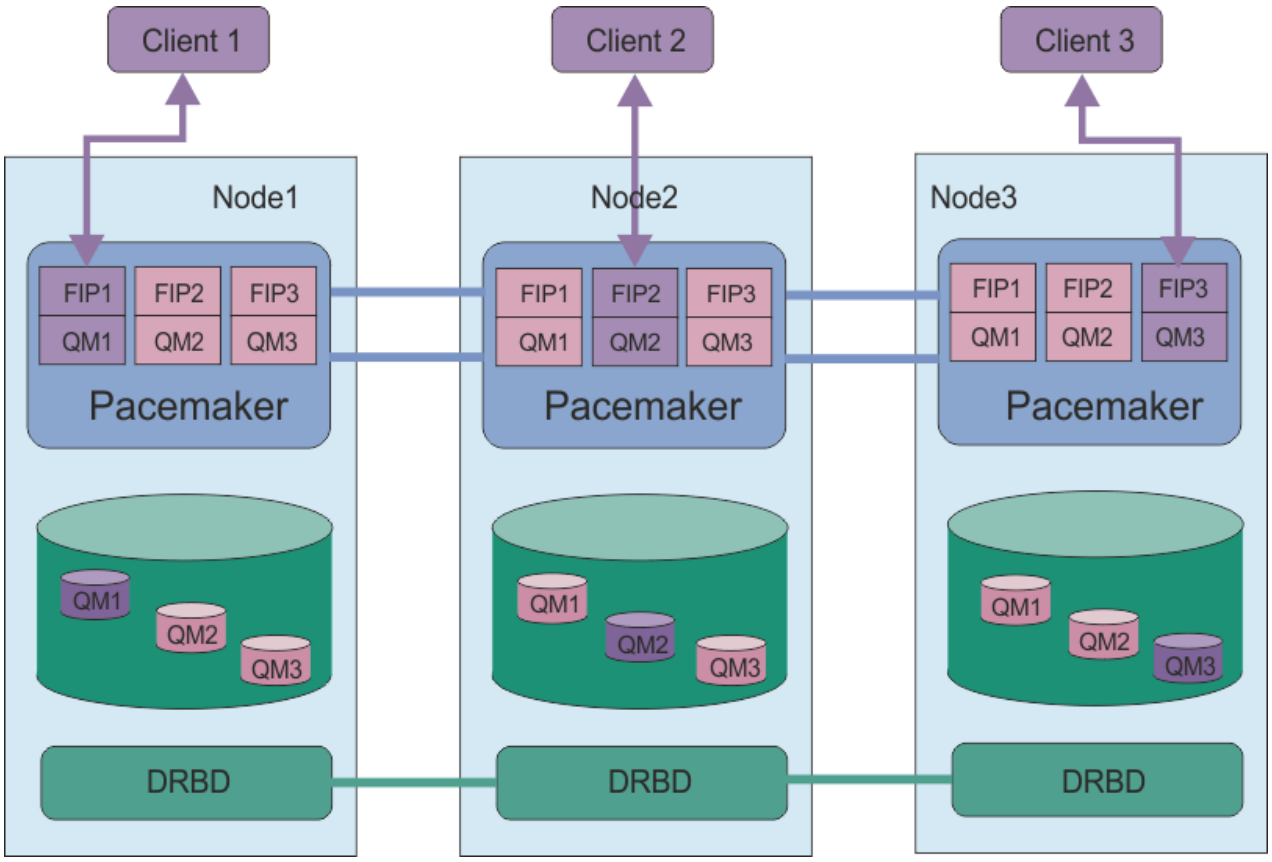


Figura 78. Ejemplo de grupo HA con tres RDQM

En la siguiente figura, Node3 ha fallado, los enlaces de Pacemaker se han perdido y el gestor de colas QM3 se ejecuta en Node2 en su lugar.

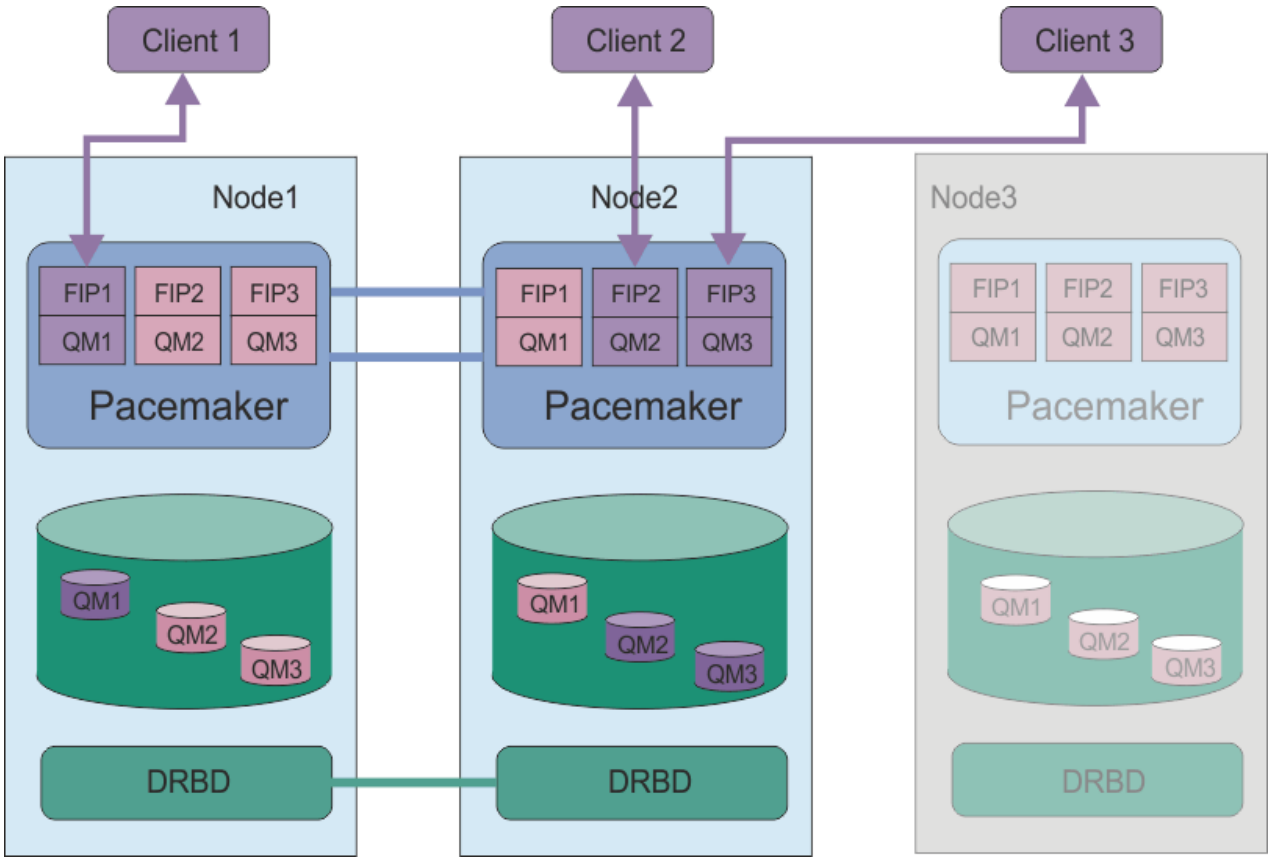


Figura 79. Ejemplo tras fallo de Node3

Nota: Cuando los gestores de colas realizan la migración tras error a otro nodo, conservan el estado que tenían en la migración tras error. Los gestores de colas que se estaban ejecutando se inician, los gestores de colas que se habían detenido permanecen detenidos.

Tareas relacionadas

Instalación de RDQM (gestores de colas de datos duplicados)

Aplicación de actualizaciones de nivel de mantenimiento para RDQM

Migración de gestores de colas de datos duplicados

Resolución de problemas de configuraciones de RDQM

Linux *Requisitos de la solución de HA de RDQM*

Hay que cumplir una serie de requisitos para configurar el grupo de alta disponibilidad (HA) del RDQM.

Requisitos del sistema

Para configurar el grupo de HA de RDQM, hay que completar algunas tareas de configuración en cada uno de los tres servidores que van a formar parte del grupo HA.

- Cada nodo requiere un grupo de volúmenes denominado `drbdpool`. El almacenamiento de cada gestor de colas de datos replicados se asigna como un volumen lógico aparte por gestor de colas de este grupo de volúmenes. Para obtener el mejor rendimiento, este grupo de volúmenes tiene que constar de uno o varios volúmenes físicos que se correspondan con unidades de disco internas (preferiblemente SSD). Puede crear `drbdpool` antes o después de haber instalado la solución de HA de RDQM, pero debe crear `drbdpool` antes de crear realmente cualquier RDQM. Consulte la configuración del grupo de volúmenes utilizando el mandato **vgs**. La salida debe ser similar a la siguiente:

```
VG       #PV #LV #SN Attr   VSize   VFree
drbdpool   1   9   0 wz--n- <16.00g <7.00g
rhel       1   2   0 wz--n- <15.00g     0
```

En particular, compruebe que no hay ningún carácter c en la sexta columna de los atributos (es decir, `wz--nc`). La c indica que la agrupación en clúster está habilitada y, si es así, debe suprimir el grupo de volúmenes y volver a crearlo sin agrupación en clúster.

- Después de haber creado el grupo de volúmenes `drbdpool`, no haga nada más con él. IBM MQ gestiona los volúmenes lógicos creados en `drbdpool` y cómo y dónde se montan.
- Cada nodo requiere hasta tres interfaces que se usan configurar el soporte RDQM:
  - Una interfaz primaria de Pacemaker para supervisar el grupo HA.
  - Una interfaz alternativa de Pacemaker para supervisar el grupo HA.
  - Una interfaz para la réplica de datos síncrona, conocida como interfaz de réplica. Esta ha de tener suficiente ancho de banda para soportar los requisitos de réplica de la carga esperada de todos los gestores de colas de datos replicados que ejecutan en el grupo HA.

  Se puede configurar el grupo HA para que se use la misma dirección IP en las tres interfaces, una dirección IP independiente en cada una de ellas o la misma dirección IP en la primaria y en la secundaria, y otra dirección IP aparte en la interfaz de réplica.

  Para maximizar la tolerancia a errores, estas interfaces han de ser tarjetas de interfaz de red (Network Interface Cards, NIC).
- DRBD requiere que cada nodo del grupo de alta disponibilidad tenga un nombre de host de Internet válido (el valor devuelto por `uname -n`), tal como se define en RFC 952 modificado por RFC 1123.
- Si hay un cortafuegos entre los nodos del grupo HA, tendrá que permitir el tráfico entre los nodos en un rango de puertos. Se proporciona un script de ejemplo, `/opt/mqm/samp/rdqm/firewalld/configure.sh`, que abre los puertos necesarios si está ejecutando el cortafuegos estándar en RHEL. Debe ejecutar el script como `root`. Si está usando algún otro cortafuegos, examine las definiciones de servicio `/usr/lib/firewalld/services/rdqm*` para ver qué puertos hay que abrir. El script añade las siguientes reglas de servicio firewallD permanentes para DRBD, Pacemaker e IBM MQ:
  - `MQ_INSTALLATION_PATH/samp/rdqm/firewalld/services/rdqm-drbd.xml` permite los puertos TCP 7000-7100.
  - `MQ_INSTALLATION_PATH/samp/rdqm/firewalld/services/rdqm-pacemaker.xml` permite puertos UDP 5404-5407
  - `MQ_INSTALLATION_PATH/samp/rdqm/firewalld/services/rdqm-mq.xml` permite el puerto TCP 1414 (debe editar el script si necesita un puerto diferente)
- Si el sistema utiliza SELinux en modalidad de imposición, es posible que tenga que ejecutar el mandato siguiente:

  ```
  semanage permissive -a drbd_t
  ```

  Si ha instalado el paquete `drbd-selinux` , no es necesario que ejecute **semanage**. Debe tener este paquete instalado en cada nodo o ejecutar **semanage** en cada nodo.

Requisitos de red

Se recomienda que localice los tres nodos del grupo de HA de RDQM en el mismo centro de datos.

Si opta por localizar los nodos en distintos centros de datos, tenga en cuenta las limitaciones siguientes:

- El rendimiento disminuye rápidamente con una mayor latencia entre centros de datos. Aunque IBM dará soporte a una latencia de hasta 5 ms, es posible que encuentre que el rendimiento de la aplicación no puede tolerar más de 1 o 2 ms de latencia.
- Los datos enviados mediante el enlace de réplica no está sujetos a ningún otro cifrado aparte del que ya se pueda aplicar al utilizar IBM MQ AMS.

Opcionalmente, puede configurar una dirección IP flotante para permitir que un cliente utilice la misma dirección IP para un gestor de colas de datos replicado (RDQM), independientemente de qué nodo del grupo HA se esté ejecutando. La dirección flotante vincula con una interfaz física con nombre en el nodo primario del RDQM. Si el RDQM migra tras error y pasa a ser primario un nodo diferente, la dirección IP

flotante está vinculada a una interfaz del mismo nombre en el nuevo primario. Las interfaces físicas en los tres nodos deben tener el mismo nombre y pertenecer a la misma subred que la dirección IP flotante.

Requisitos de usuario para configurar el clúster

Puede configurar el grupo de alta disponibilidad de RDQM como usuario `root`. Si no desea configurar como `root`, configure como usuario en el grupo `mqm` en su lugar. Para que un usuario del grupo `mqm` configure el clúster RDQM, debe cumplir los requisitos siguientes:

- El usuario de `mqm` debe poder utilizar sudo para ejecutar mandatos en cada uno de los tres servidores que componen el grupo de HA de RDQM.
- Si el usuario de `mqm` puede utilizar SSH sin una contraseña para ejecutar mandatos en cada uno de los tres servidores que componen el grupo de HA de RDQM, el usuario debe ejecutar mandatos en únicamente uno de los servidores.
- El usuario de `mqm` debe tener el mismo UID en los tres servidores.
- El grupo `mqm` debe tener el mismo GID en los tres servidores.

Debe configurar sudo para que el usuario `mqm` pueda ejecutar los mandatos siguientes con autorización root:

```
/opt/mqm/bin/crtmqm
/opt/mqm/bin/dltmqm
/opt/mqm/bin/rdqmadm
/opt/mqm/bin/rdqmstatus
```

Requisitos de usuario para trabajar con gestores de colas

Para crear, suprimir o configurar gestores de colas de datos replicados (RDQM), debe utilizar un ID de usuario que pertenezca a los grupos `mqm` y `haclient` (el grupo `haclient` se crea durante la instalación de Pacemaker).

Linux *Configuración del acceso SSH sin contraseña y sudo*

Puede configurar SSH sin contraseña y acceso sudo para que sólo necesite emitir mandatos de configuración en un nodo del grupo HA. (La configuración de este acceso es opcional, o bien puede ejecutar mandatos en cada nodo.)

Acerca de esta tarea

Para configurar un SSH sin contraseña, hay que configurar el id `mqm` en cada nodo y luego generar una clave para dicho usuario en cada nodo. Luego se distribuyen las claves a los otros nodos y se prueba la conexión para añadir cada nodo a la lista de hosts conocidos. Por último, bloquee el ID de `mqm` .

Nota: Las instrucciones presuponen que se está definiendo un grupo HA con interfaces primaria, alternativa y de réplica distintas, y que, por tanto, se define un acceso SSH sin contraseña a través de las interfaces primaria y alternativa. Si tiene previsto configurar un sistema con una única dirección IP, se define un acceso SSH sin contraseña a través de esa interfaz única. Si tiene previsto configurar un sistema con dos direcciones IP para HA_Primary y HA_Replication, el ssh debe estar configurado para la dirección HA_Primary.

A continuación, puede crear el acceso sudo para el ID de `mqm` en cada nodo.

Procedimiento

1. Para configurar SSH sin contraseña:
   a) En cada uno de los tres nodos, siga los pasos siguientes para configurar el usuario `mqm` y generar una clave SSH:

i) Cambie el directorio de inicio de `mqm` a `/home/mqm`:

```
usermod -d /home/mqm mqm
```

ii) Cree el directorio `/home/mqm`:

```
mkhomedir_helper mqm
```

iii) Añada la contraseña de `mqm`:

```
passwd mqm
```

iv) Ejecute el shell interactivo con `mqm`:

```
su mqm
```

v) Genere la clave de autenticación de `mqm`:

```
ssh-keygen -t rsa -f /home/mqm/.ssh/id_rsa -N ''
```

b) En cada uno de los tres nodos, siga los pasos siguientes para añadir la clave de dicho nodo a los otros dos nodos y probar las conexiones de las direcciones primaria y (si se usa) secundaria de cada nodo:

i) Añada la clave a los nodos remotos:

```
ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node1_primary_address
ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node1_alternate_address
ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node2_primary_address
ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node2_alternate_address
```

ii) Compruebe el SSH sin contraseña y actualice known_hosts en los nodos remotos:

```
ssh remote_node1_primary_address uname -n
ssh remote_node1_alternate_address uname -n
ssh remote_node2_primary_address uname -n
ssh remote_node2_alternate_address uname -n
```

En cada conexión, se le solicitará que confirme que desea continuar. Confirme cada una para actualizar known_hosts. Tiene que completar esto antes de intentar configurar el grupo HA usando SSH sin contraseña.

iii) Salga del shell interactivo como `mqm`:

```
exit
```

c) En cada nodo, con root, siga los pasos siguientes para eliminar la contraseña de `mqm` y bloquear el id:

i) Elimine la contraseña de `mqm`:

```
passwd -d mqm
```

ii) Bloquee `mqm`:

```
passwd -l mqm
```

2. En cada nodo, como root, configure el acceso sudo para el usuario `mqm` creando el archivo `/etc/sudoers.d/mqm` que contiene el texto siguiente:

```
mqm ALL=(root) NOPASSWD: /opt/mqm/bin/crtmqm, /opt/mqm/bin/dltmqm, /opt/mqm/bin/
rdqmadm, /opt/mqm/bin/rdqmstatus, /opt/mqm/bin/rdqmdr
```

Linux *Definición del clúster de Pacemaker (grupo HA)*

El grupo HA es un clúster Pacemaker. El clúster Pacemaker se define editando el archivo `/var/mqm/rdqm.ini` y ejecutando el mandato **rdqmadm**.

Acerca de esta tarea

Consulte https://clusterlabs.org/pacemaker/ para obtener información sobre Pacemaker. Puede crear el clúster Pacemaker con un usuario del grupo `mqm` si el usuario `mqm` puede utilizar `sudo`. Si el usuario también puede hacer SSH a cada servidor sin contraseña, entonces solo es necesario editar el archivo `rdqm.ini` y ejecutar **rdqmadm** en uno de los servidores para crear el clúster Pacemaker. De lo contrario, hay que crear el archivo y ejecutar el mandato con `root` en cada uno de los servidores que vayan a ser nodos.

El archivo `rdqm.ini` proporciona las direcciones IP que utiliza RDQM para los nodos del clúster Pacemaker . Debe dar el nombre de cada nodo, que debe ser el nombre de host tal como lo devuelve el mandato **uname -n** .

Un grupo HA RDQM se puede configurar para utilizar una, dos o tres direcciones IP:

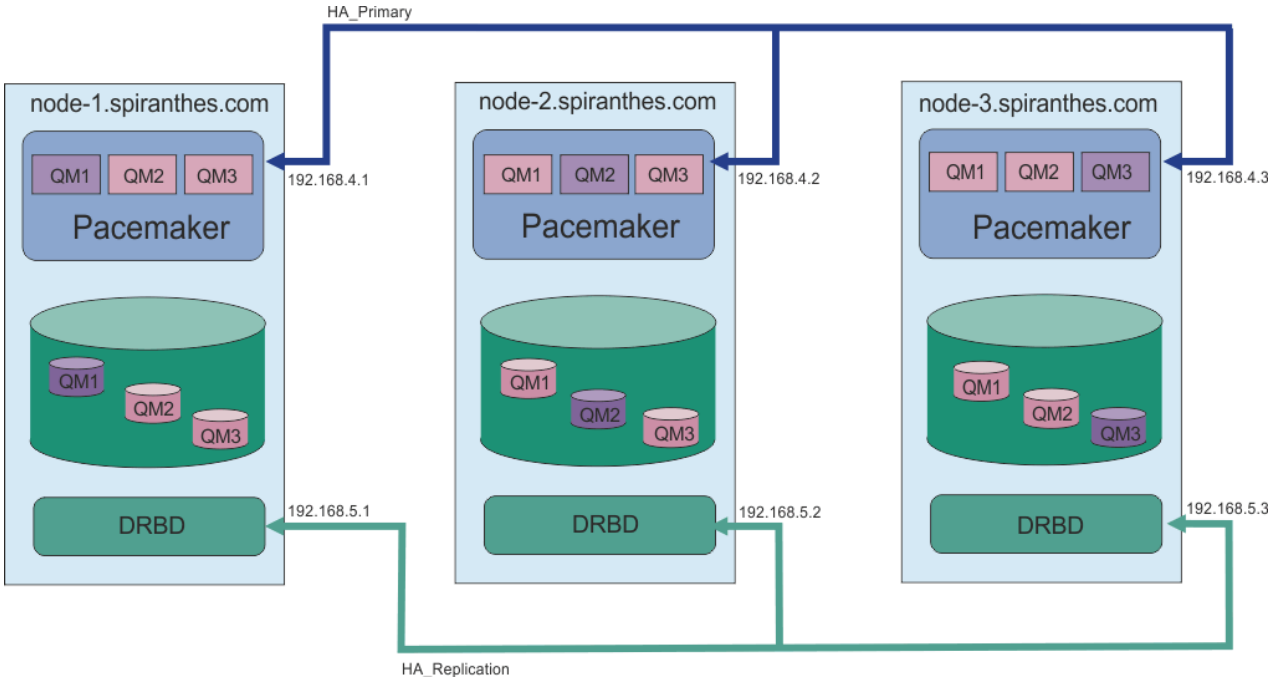
- Una dirección IP: las pulsaciones y la réplica comparten el mismo enlace
- Dos direcciones IP: las pulsaciones y la réplica utilizan enlaces separados
- Tres direcciones IP: un enlace para la réplica y dos enlaces separados para pulsaciones

Estas opciones se proporcionan para dar soporte a diferentes patrones de despliegue para RDQM. Las distintas opciones se pueden utilizar para maximizar la resiliencia de la solución RDQM basándose en el entorno que se utiliza. Las configuraciones que utilizan dos o tres direcciones IP están pensadas principalmente para despliegues en los que es necesario un control granular sobre qué red física enlaza las pulsaciones y el tráfico de réplica para configurar la redundancia para la conectividad entre nodos. De forma alternativa, se puede implementar una conectividad altamente disponible y resiliente en la capa de red, por ejemplo, utilizando la agregación de enlaces. Con la agregación de enlaces, se utilizan varios enlaces de red física para proporcionar un único enlace lógico que puede seguir funcionando si fallan enlaces físicos individuales. Si RDQM se despliega en un entorno donde la conectividad de red está virtualizada, y/o donde la conectividad resiliente se implementa en la capa de red, normalmente es preferible utilizar una única dirección IP para las pulsaciones y la réplica.

El ejemplo siguiente ilustra el uso de dos direcciones IP. El archivo `rdqm.ini` tiene un campo `HA_Primary` y un campo `HA_Replication` para cada nodo, pero ningún campo `HA_Alternate` :

```
Node:
  Name=rdqm-node-1.spiranthes.com
  HA_Primary=192.168.4.1
  HA_Replication=192.168.5.1
Node:
  Name=rdqm-node-2.spiranthes.com
  HA_Primary=192.168.4.2
  HA_Replication=192.168.5.2
Node:
  Name=rdqm-node-3.spiranthes.com
  HA_Primary=192.168.4.3
  HA_Replication=192.168.5.3
```

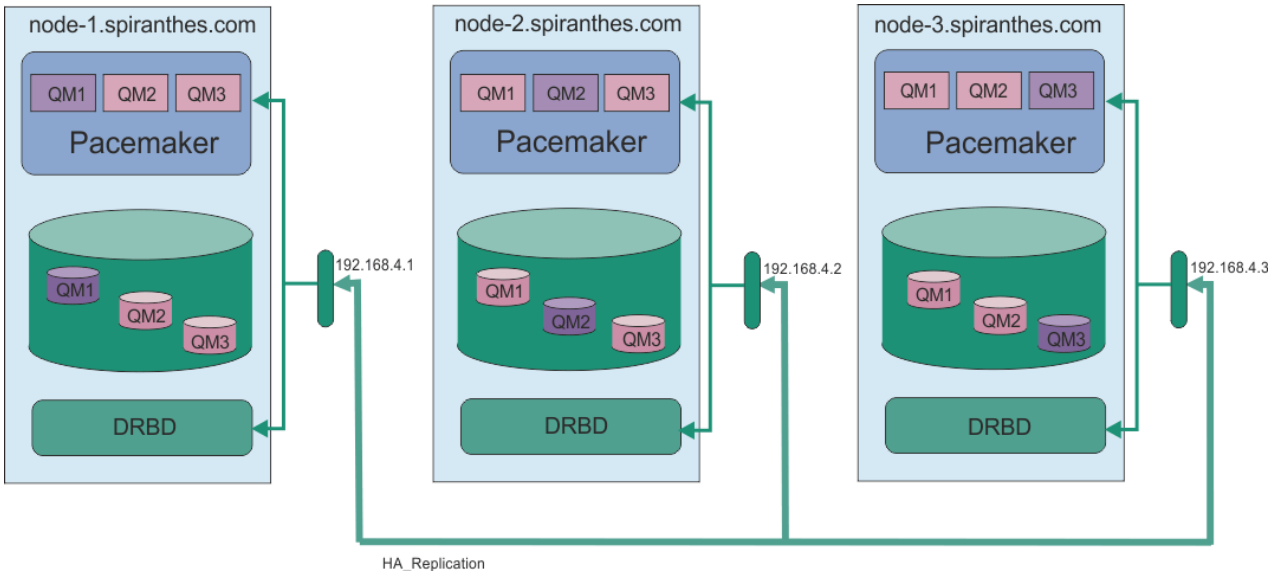
El diagrama siguiente ilustra esta configuración:



El siguiente archivo de ejemplo muestra la configuración de un clúster Pacemaker de ejemplo que utiliza la interfaz `HA_Replication` para la supervisión. En este caso, solo se especifica la interfaz `HA_Replication`:

```
Node:
  Name=rdqm-node-1.spiranthes.com
  HA_Replication=192.168.4.1
Node:
  Name=rdqm-node-2.spiranthes.com
  HA_Replication=192.168.4.2
Node:
  Name=rdqm-node-3.spiranthes.com
  HA_Replication=192.168.4.3
```

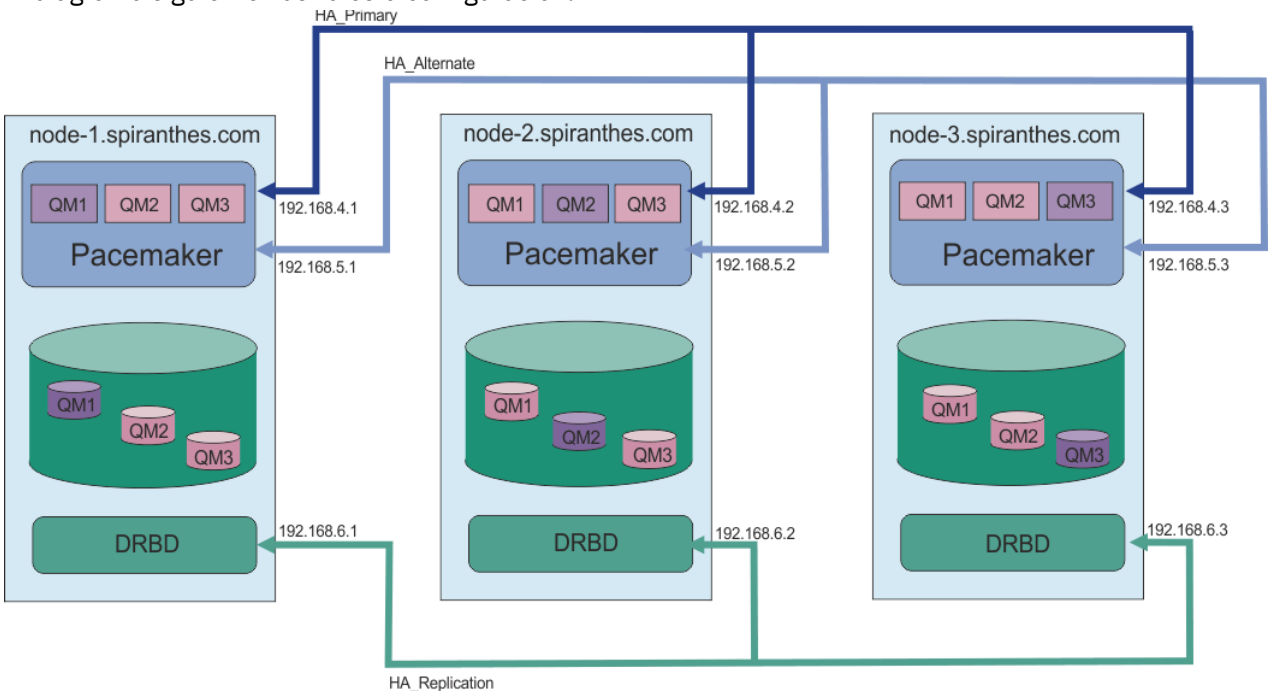
El diagrama siguiente ilustra esta configuración:



El siguiente archivo de ejemplo muestra la configuración de un clúster Pacemaker de ejemplo que utiliza una dirección IP independiente para cada interfaz:

```
Node:
  Name=rdqm-node-1.spiranthes.com
  HA_Primary=192.168.4.1
  HA_Alternate=192.168.5.1
  HA_Replication=192.168.6.1
Node:
  Name=rdqm-node-2.spiranthes.com
  HA_Primary=192.168.4.2
  HA_Alternate=192.168.5.2
  HA_Replication=192.168.6.2
Node:
  Name=rdqm-node-3.spiranthes.com
  HA_Primary=192.168.4.3
  HA_Alternate=192.168.5.3
  HA_Replication=192.168.6.3
```

El diagrama siguiente ilustra esta configuración:



El orden en el que especifique los nodos debe ser el mismo en todos los archivos `rdqm.ini` de la configuración. Los tres nodos deben tener una vista común con respecto a cuál es Nodo1, cuál es Nodo2, etc.

Procedimiento

- Para definir el clúster Pacemaker con el usuario `root`:
  a) Edite el archivo `/var/mqm/rdqm.ini` en uno de los tres servidores para que el archivo defina el clúster.
  b) Copie el archivo en los otros dos servidores que vayan a ser nodos del clúster Pacemaker.
  c) Ejecute el mandato siguiente como `root` en cada uno de los tres servidores:

  ```
  rdqmadm -c
  ```

- Para definir el clúster Pacemaker como un usuario del grupo `mqm` en cada nodo:
  a) Asegúrese de que el usuario `mqm` puede utilizar **sudo** para ejecutar los mandatos.
  b) Edite el archivo `/var/mqm/rdqm.ini` en uno de los tres servidores para que el archivo defina el clúster de Pacemaker.
  c) Copie `/var/mqm/rdqm.ini` en los otros dos servidores que serán nodos en el clúster de Pacemaker.

d) Ejecute el mandato siguiente en cada servidor:

```
rdqmadm -c
```

- Para definir el clúster Pacemaker como un usuario del grupo `mqm` desde un nodo:
  a) Asegúrese de que el usuario `mqm` puede utilizar **sudo** para ejecutar mandatos y puede conectarse a cada servidor utilizando SSH sin contraseña.
  b) Edite el archivo `/var/mqm/rdqm.ini` en uno de los tres servidores para que el archivo defina el clúster de Pacemaker.
  c) Ejecute el siguiente mandato:

```
rdqmadm -c
```

Referencia relacionada
rdqmadm (administrar un clúster de gestores de colas de datos replicados)

Linux *Supresión del clúster de Pacemaker (grupo HA)*
El grupo HA es un clúster Pacemaker. Se puede borrar una configuración de clúster Pacemaker ejecutando el mandato **rdqmadm** con la opción `-u`.

Acerca de esta tarea

No se puede borrar la configuración del clúster Pacemaker si aún existe algún gestor de colas de datos replicado en cualquiera de los nodos.

Procedimiento

- Para borrar la configuración del clúster Pacemaker, ejecute el mandato siguiente en cualquiera de los nodos:

```
rdqmadm -u
```

Referencia relacionada
rdqmadm (administrar un clúster de gestores de colas de datos replicados)

Linux ***Creación de un RDQM de HA***
El mandato **crtmqm** se utiliza para crear un gestor de colas de datos replicados de alta disponibilidad (RDQM).

Acerca de esta tarea

Puede crear un gestor de colas de datos replicados de alta disponibilidad (RDQM) como usuario del grupo `mqm` si el usuario mqm puede utilizar sudo. Si el usuario también puede hacer SSH en cada nodo sin contraseña, entonces solo se necesita ejecutar el mandato de creación de RDQM en un nodo para crearlo en los tres nodos. De lo contrario hay que ser `root` para crear un RDQM los mandatos se tienen que ejecutar en los tres nodos.

Nota: Hay un límite absoluto de 129 gestores de colas en un grupo HA. Si intenta crear más de esto, el intento fallará. En la práctica, la adición de más de 50 gestores de colas a un grupo HA puede encontrar problemas de tiempo de espera.

Los puntos siguientes proporcionan algunas directrices sobre el dimensionamiento del sistema de archivos del gestor de colas:

1. Al crear un gestor de colas RDQM, se asigna un sistema de archivos para almacenar datos y registros del gestor de colas. Es importante dimensionar este sistema de archivos de forma adecuada para que el gestor de colas pueda registrar la actividad en curso en sus registros y almacenar los mensajes de aplicación en las colas. Al dimensionar el sistema de archivos, tenga en cuenta los requisitos máximos de mensajería, el crecimiento futuro de la carga de trabajo y las paradas de las aplicaciones que

pueden hacer que los mensajes se acumulen en las colas. Para obtener instrucciones sobre cómo calcular el tamaño del registro de recuperación del gestor de colas, consulte "¿Qué tamaño debe tener el sistema de archivos de registro?" en la página 685. Al calcular los requisitos de almacenamiento para los mensajes de aplicación, es necesario tener en cuenta el tamaño y el número de mensajes, además de su cabecera MQMD y cualquier propiedad de mensaje que tengan.

2. Los sistemas de archivos del gestor de colas RDQM no se pueden redimensionar dinámicamente. Debe realizar una copia de seguridad y, a continuación, restaurar un gestor de colas RDQM con un sistema de archivos más grande si es necesario, consulte "Redimensionar el sistema de archivos para un gestor de colas RDQM HA" en la página 611.
3. Puede limitar el tamaño de colas individuales en disco utilizando atributos de cola local, como MAXDEPTH y MAXFSIZE. Consulte Modificación de archivos de cola de IBM MQ.
4. Debe supervisar el uso de disco en curso y responder adecuadamente si el uso de disco aumenta antes de que el uso del sistema de archivos pase a ser crítico. El uso del sistema de archivos se puede supervisar utilizando las prestaciones de plataforma/sistema operativo o suscribiéndose a las métricas publicadas en los temas del sistema IBM MQ que se describen en Métricas publicadas en los temas del sistema.

Procedimiento

- Para crear un RDQM como un usuario del grupo `mqm`:
  a) Asegúrese de que el usuario `mqm` pueda utilizar **sudo** para ejecutar mandatos y que pueda conectarse a cada servidor utilizando SSH sin contraseña.
  b) Escriba el mandato siguiente:

```
crtmqm -sx [-fs FilesystemSize] qmname
```

  donde *nombreGC* es el nombre del gestor de colas de datos replicados. De forma opcional, se puede especificar el tamaño del sistema de archivos del gestor de colas (es decir, el tamaño del volumen lógico que se crea en el grupo de volúmenes drbdpool).

  El mandato intenta utilizar SSH para conectarse con los otros nodos del clúster como un usuario de `mqm`. Si la conexión es satisfactoria, se crean las instancias secundarias en los nodos. De lo contrario, hay que crear las instancias secundarias y luego ejecutar el mandato **`crtmqm -sx`** (tal y como se describe para el usuario `root`).
- Para crear un RDQM como usuario `root`:
  a) Ejecute el mandato siguiente en cada uno de los nodos que van a alojar las instancias secundarias del RDQM:

```
crtmqm -sxs [-fs FilesystemSize] qmname
```

  donde *nombreGC* es el nombre del gestor de colas de datos replicados. De forma opcional, se puede especificar el tamaño del sistema de archivos del gestor de colas (es decir, el tamaño del volumen lógico que se crea en el grupo de volúmenes drbdpool). Hay que especificar el mismo tamaño de sistema de archivos para el RDQM en los tres nodos del grupo HA. El tamaño es un valor numérico, que se especifica en GB. Puede especificar un valor en MB especificando el valor seguido del carácter `M`

  El mandato crea una instancia secundaria del RDQM.
  b) En el nodo restante, ejecute el mandato siguiente:

```
crtmqm -sx [-fs FilesystemSize] qmname
```

  donde *nombreGC* es el nombre del gestor de colas de datos replicados. De forma opcional, se puede especificar el tamaño del sistema de archivos del gestor de colas. El tamaño es un valor numérico, que se especifica en GB. Puede especificar un valor en MB especificando el valor seguido del carácter `M`.

El mandato determina si la instancia secundaria del gestor de colas existe en los otros dos nodos. Si existen los secundarios, el mandato crea e inicia el gestor de colas primario. Si los secundarios no existen, se le indicará que ejecute el mandato **crtmqm -sxs** en cada uno de los nodos.

Aparte de los argumentos DataPath (**-md**) y LogPath (**-ld**), todos los argumentos que son válidos para crear un gestor de colas Linux estándar también son válidos para un gestor de colas de datos replicados primario.

Nota: Cuando se crea un RDQM, se asigna el siguiente número de puerto libre por encima de 7000 para el enlace de réplica. Si se descubre que el puerto elegido es utilizado por otra aplicación, el mandato **crtmqm** falla con el error AMQ6543 y dicho puerto se añade a una lista de exclusión. Debe suprimir las instancias secundarias del gestor de colas y, a continuación, ejecutar de nuevo el mandato **crtmqm**.

Referencia relacionada

crtmqm

Linux *Supresión de un RDQM de HA*

El mandato **dltmqm** se utiliza para suprimir un gestor de colas de datos replicados de alta disponibilidad (RDQM).

Acerca de esta tarea

Hay que ejecutar el mandato para borrar el RDQM en el nodo primario del RDQM. Antes hay que parar el RDQM. El mandato se puede ejecutar como usuario mqm si dicho usuario tiene los privilegios sudo necesarios. De lo contrario, hay que ejecutar el mandato como root. Una vez borrados los recursos asociados al gestor de colas primario, el mandato intenta borrar los gestores de colas secundarios conectándose mediante SSH con los otros nodos. Si dicho borrado falla, habrá que ejecutar `dltmqm` manualmente en los demás nodos para completar el proceso. En un nodo secundario, el mandato falla si el gestor de colas primario aún no se ha borrado.

Procedimiento

- Para borrar un RDQM, ejecute el mandato siguiente:

```
dltmqm RDQM_name
```

Referencia relacionada

dltmqm

MQ Adv. Linux *Migración de un gestor de colas para que se convierta en un gestor de colas HA RDQM*

Puede migrar un gestor de colas existente para que se convierta en un gestor de colas de datos replicados (RDQM) de alta disponibilidad (HA) haciendo una copia de seguridad de sus datos persistentes y, después, restaurando los datos en un nuevo gestor de colas RDQM recién creado que tenga el mismo nombre.

Acerca de esta tarea

Los gestores de colas de datos replicados HA requieren un volumen lógico dedicado (sistema de archivos) y la configuración de la réplica de disco y el control de HA. Estos componentes solo se configuran cuando se crea un nuevo gestor de colas. Un gestor de colas existente se puede migrar para utilizar el RDQM haciendo una copia de seguridad de sus datos persistentes y, después, restaurando los datos en un gestor de colas RDQM recién creado que tenga el mismo nombre. Este procedimiento conserva la configuración del gestor de colas, el estado y los mensajes persistentes en el momento en que se creó la copia de seguridad.

Nota: Solo puede migrar un gestor de colas de una versión de IBM MQ que sea igual o inferior a la versión en la que se ha instalado el RDQM. El sistema operativo y la arquitectura también debe ser los mismos.

De lo contrario, debe crear un nuevo gestor de colas en la plataforma de destino, consulte Trasladar un gestor de colas a un sistema operativo diferente.

Antes de migrar un gestor de colas, debe cumplir las condiciones siguientes:

- Evalúe los requisitos de alta disponibilidad y consulte "Alta disponibilidad en RDQM" en la página 595.
- Revise las aplicaciones y los gestores de colas que se conectan al gestor de colas. Considere los cambios necesarios para direccionar las conexiones al nodo RDQM donde se está ejecutando el gestor de colas. Por ejemplo, si configura la alta disponibilidad de RDQM, podría tener en cuenta utilizar una dirección IP flotante, consulte "Creación y borrado de una dirección IP flotante" en la página 614.
- Suministre, o identifique, los nodos RDQM existentes para la configuración que seleccione. Para obtener más información sobre los requisitos del sistema para el RDQM, consulte "Requisitos de la solución de HA de RDQM" en la página 598.
- Instale IBM MQ Advanced, que incluye la característica RDQM, en cada nodo.
- Configure la configuración del grupo HA de RDQM, consulte "Definición del clúster de Pacemaker (grupo HA)" en la página 601.
- Si lo desea, verifique la configuración de RDQM utilizando un gestor de colas de prueba, que después se puede suprimir. Se recomienda probar la configuración para identificar y resolver los problemas antes de migrar el gestor de colas.
- Revise la configuración de seguridad para el gestor de colas y, a continuación, duplique los grupos y usuarios locales necesarios en cada nodo RDQM.
- Revise el gestor de colas y la configuración de canal para determinar si se utilizan salidas de API, salidas de canal o salidas de conversión de datos. Instale las salidas necesarias en cada nodo RDQM.
- Revise los servicios del gestor de colas que se han definido y, a continuación, instale y configure los procesos necesarios en cada nodo RDQM.

Procedimiento

1. Haga una copia de seguridad del gestor de colas existente:
   a) Detenga el gestor de colas existentes emitiendo un mandato de conclusión en espera `endmqm -w`, o un mandato de conclusión inmediata `endmqm -i`. Este paso es importante para garantizar que los datos de la copia de seguridad son coherentes.
   b) Determine la ubicación del directorio de datos del gestor de colas visualizando el archivo de configuración de IBM MQ, `mqs.ini`. En Linux, este archivo se encuentra en el directorio `/var/mqm`. Para obtener más información sobre `mqs.ini`, consulte "Archivo de configuración de IBM MQ, mqs.ini" en la página 96.

      Localice la stanza `QueueManager` para el gestor de colas en el archivo. Si la stanza contiene una clave llamada `DataPath`, su valor es el directorio de datos del gestor de colas. Si la clave no existe, el directorio de datos del gestor de colas se puede determinar utilizando los valores de las claves `Prefix` y `Directory`. El directorio de datos del gestor de colas es una concatenación de estos valores, con el formato *prefijo*/qmgrs/*directorio*. Para obtener más información sobre la stanza QueueManager, consulte "Stanza QueueManager del archivo mqs.ini" en la página 107.
   c) Cree una copia de seguridad del directorio de datos de gestor de colas. En Linux, puede hacerlo utilizando el mandato **tar**. Por ejemplo, para hacer una copia de seguridad del directorio de datos para un gestor de colas, puede utilizar el mandato siguiente. Observe el último parámetro del mandato, que es un único punto:

      ```
      tar -cvzf qm-data.tar.gz -C queue_manager_data_dir .
      ```

   d) Determine la ubicación del directorio de registro del gestor de colas visualizando el archivo de configuración del gestor de colas de IBM MQ `qm.ini`. Este archivo se encuentra en el directorio de datos del gestor de colas. Para obtener más información sobre el archivo, consulte "Archivos de configuración de gestores de colas, qm.ini" en la página 109.

El directorio de registro del gestor de colas se define como el valor de la clave `LogPath` en la stanza `Log`. Para obtener información sobre la stanza, consulte "Stanza de registro del archivo qm.ini" en la página 145.

e) Cree una copia de seguridad del directorio de registro del gestor de colas. En Linux, puede hacer esto utilizando el mandato tar. Por ejemplo, para hacer una copia de seguridad del directorio de registro para un gestor de colas, puede utilizar el mandato siguiente. Observe el último parámetro del mandato, que es un único punto:

```
tar -cvzf qm-log.tar.gz -C queue_manager_log_dir .
```

f) Cree una copia de seguridad de los repositorios de certificados utilizados por el gestor de colas, si no se encuentran en el directorio de datos del gestor de colas. Asegúrese de que se ha hecho una copia de seguridad de ambos archivos, el archivo de base de datos de claves y el archivo de ocultación de contraseña. Para obtener información sobre el repositorio de claves del gestor de colas, consulte El repositorio de claves SSL/TLS y Ubicación del repositorio de claves para un gestor de colas. Para obtener más información sobre cómo localizar el almacén de claves AMS, si el gestor de colas se ha configurado para utilizar la intercepción del agente de canal de mensajes (MCA) AMS, consulte Intercepción del agente de canal de mensajes (MCA).

g) El gestor de colas existente ya no es necesario, así que se puede suprimir. Sin embargo, siempre que sea posible, solo debe suprimir el gestor de colas existente, después de que se haya restaurado correctamente en el sistema de destino. El aplazamiento de la supresión garantiza que el gestor de colas se puede reiniciar si el proceso de migración no se completa correctamente.

Nota: Si aplaza la supresión del gestor de colas existente, no lo reinicie. Es importante que el gestor de colas permanezca finalizado porque los cambios adicionales en si configuración o estado se pierden durante la migración.

2. Prepare el nodo RDQM primario:

a) Cree un nuevo gestor de colas RDQM con el mismo nombre que el gestor de colas del que ha hecho una copia de seguridad. Asegúrese de que el sistema de archivos asignado para el gestor de colas RDQM por **crtmqm** es lo suficientemente grande para que contenga los datos, los registros primarios y los registros secundarios para el gestor de colas existente, además de algún espacio adicional para una futura ampliación. Para obtener información sobre cómo crear un gestor de colas RDQM, consulte "Creación de un RDQM de HA" en la página 605.

b) Determine el nodo RDQM primario para el gestor de colas. Para obtener más información sobre cómo determinar el nodo primario, consulte rdqmstatus (mostrar estado RDQM).

c) En el nodo RDQM primario, si el gestor de colas RDQM se inicia, deténgalo utilizando el mandato `endmqm -w` o `endmqm -i`.

d) En el nodo RDQM primario, determine la ubicación de los directorios de datos y registro para el gestor de colas RDQM (utilice los métodos descritos en los pasos 1b y 1d).

e) En el nodo RDQM primario, suprima el contenido de los directorios de datos y registros del gestor de colas RDQM, pero no los propios directorios.

3. Restaure el gestor de colas en el nodo RDQM primario:

a) Copie las copias de seguridad de los directorios de datos y registro del gestor de colas en el nodo RDQM primario, además de las copias de seguridad independientes de los repositorios de certificados utilizados por el gestor de colas.

b) Restaure la copia de seguridad del directorio de datos del gestor de colas en el directorio de datos vacío para el nuevo gestor de colas RDQM, asegurándose de que se conservan los permisos y la propiedad de archivos. Si la copia de seguridad se ha creado utilizando el mandato tar de ejemplo en el paso 1c, el usuario root puede utilizar el mandato siguiente para restaurarla:

```
tar -xvzpf qm-data.tar.gz -C queue_manager_data_dir
```

c) Restaure la copia de seguridad del directorio de registro del gestor de colas en el directorio de registro vacío para el nuevo gestor de colas RDQM, asegurándose de que se conserven los permisos

y la propiedad de archivos. Si la copia de seguridad se ha creado utilizando el mandato tar de ejemplo en el paso 1e, el usuario root puede utilizar el mandato siguiente para restaurarla:

```
tar -xvzpf qm-log.tar.gz -C queue_manager_log_dir
```

d) Edite el archivo de configuración del gestor de colas restaurado, `qm.ini`, en el directorio de datos del gestor de colas RDQM. Actualice el valor de la clave `LogPath` en la stanza `Log` para especificar el directorio de registro para el gestor de colas RDQM.

Revise otras vías de acceso de archivo que están definidas en el archivo de configuración y actualícelas, si es necesario. Por ejemplo, es posible que tenga que actualizar las vías de acceso siguientes:

- La vía de acceso de los archivos de registro de errores generados por los servicios de mensajes de diagnóstico.
- La vía de acceso para las salidas necesarias para el gestor de colas.
- La vía de acceso para los archivos de carga conmutada, si el gestor de colas es un coordinador de transacciones XA.

e) Si el gestor de colas se ha configurado para utilizar la intercepción del agente de canal de mensajes (MCA) AMS, copie el almacén de claves AMS en la nueva instalación de RDQM y, después, revise y actualice la configuración. El almacén de claves debe estar disponible en cada nodo RDQM, de modo que si no se encuentra en el sistema de archivos duplicado para el gestor de colas, en su lugar, se debe copiar en cada nodo. Para obtener más información, consulte Intercepción del agente de canal de mensajes (MCA).

f) Verifique que el gestor de colas se muestra mediante el mandato **dspmq** y que su estado se notifica como finalizado. El ejemplo siguiente muestra la salida de ejemplo para un gestor de colas RDQM HA.

```
$ dspmq -o status -o ha

QMNAME(QM1) STATUS(Ended normally) HA(Replicated)
```

g) Verifique que los datos del gestor de colas restaurado se han duplicado en los nodos RDQM secundarios utilizando el mandato **rdqmstatus** para mostrar el estado del gestor de colas. El estado de HA se debe notificar como `Normal` en cada nodo. El ejemplo siguiente muestra la salida de ejemplo para un gestor de colas RDQM HA.

```
$ rdqmstatus -m QM1
Node:                           mqhavm10-adm
Queue manager status:           Ended normally
Queue manager file system:      50MB used, 0.2GB allocated [42%]
HA role:                        Primary
HA status:                      Normal
HA control:                     Disabled
HA current location:            This node
HA preferred location:          This node
HA floating IP interface:       None
HA floating IP address:         None

Node:                           mqhavm11-adm
HA status:                      Normal

Node:                           mqhavm12-adm
HA status:                      Normal
```

h) Inicie el gestor de colas en el nodo RDQM primario.

i) Conéctese al gestor de colas y actualice el valor del atributo del gestor de colas SSLKEYR para especificar la nueva ubicación del repositorio de certificados del gestor de colas. De forma predeterminada, el valor de este atributo se establece en *queue_manager_data_directory*`/ssl/key`. El repositorio de certificados debe estar ubicado en la misma ubicación en cada nodo RDQM. Si el repositorio no se encuentra en el sistema de archivos duplicado para el gestor de colas, en su lugar, se debe copiar en cada nodo.

j) Revise las definiciones de objeto de IBM MQ para el gestor de colas y actualice el valor de los atributos del objeto que hacen referencia a los valores de red cambiados, el directorio de instalación de IBM MQ o el directorio de datos del gestor de colas, incluidos los objetos siguientes:

- Direcciones IP locales utilizadas por escuchas (atributo IPADDR)
- Direcciones IP locales utilizadas por canales (atributo LOCLADDR)
- Direcciones IP locales definidas para los canales de clúster receptor (atributo CONNAME)
- Direcciones IP locales definidas para los objetos de información de comunicación (atributo GRPADDR)
- Vías de acceso de sistema definidas para las definiciones de objeto de proceso y servicio.

k) Detenga y reinicie el gestor de colas para asegurarse de que los cambios acaben siendo efectivos.

l) Repita el paso 3j para los gestor de colas remotos. además de los valores equivalentes para las aplicaciones, que se conectan al gestor de colas migrados, incluyendo:

- Nombre de conexión de canal (atributo CONNAME)
- Reglas de autenticación de canal que restringen las conexiones de entrada del gestor de colas basándose en su dirección IP o nombre de host.
- Tablas de definición de canal de cliente (CCDT), valores de nombre de dominio (DNS), direccionamiento de red o información de conexión equivalente.

m) Realice una migración tras error gestionada del gestor de colas a cada nodo RDQM para garantizar que se ha establecido correctamente la configuración necesaria, consulte "Configuración de la ubicación preferida de un RDQM" en la página 614.

Redimensionar el sistema de archivos para un gestor de colas RDQM HA
Para redimensionar el sistema de archivos para un gestor de colas de datos replicados (RDQM) de alta disponibilidad (HA), haga una copia de seguridad de sus datos persistentes y, después, restaure los datos en un gestor de colas RDQM recién creado que tenga el mismo nombre, pero un sistema de archivo de un tamaño diferente.

Acerca de esta tarea

Los gestores de colas de datos replicados de HA requieren un volumen lógico dedicado (sistema de archivos) y la configuración de la réplica de disco y el control de HA. Estos componentes solo se configuran cuando se crea un nuevo gestor de colas. El sistema de archivos no se puede redimensionar después de que haya sido creado porque debe tener el mismo tamaño en cada nodo. Para redimensionar el sistema de archivos para un gestor de colas de datos duplicados (RDQM) existente, puede hacer una copia de seguridad de sus datos persistentes y, a continuación, restaurar los datos en un gestos de datos RDQM recién creado que tenga el mismo nombre, pero un sistema de archivos de un tamaño diferente. Este procedimiento conserva la configuración del gestor de colas, el estado y los mensajes persistentes en el momento en que se creó la copia de seguridad.

Procedimiento

1. Haga una copia de seguridad del gestor de colas RDQM existente en el nodo RDQM primario:
   a) Determine el nodo RDQM primario para el gestor de colas. Para obtener más información sobre cómo determinar el nodo primario, consulte rdqmstatus (mostrar estado RDQM).
   b) En el nodo RDQM primario, si el gestor de colas RDQM se inicia, deténgalo utilizando el mandato **endmqm -w** o **endmqm -i**.
   c) Determine la ubicación del directorio de datos del gestor de colas visualizando el archivo de configuración de IBM MQ, mqs.ini. En Linux, este archivo se encuentra en el directorio /var/mqm. Para obtener más información sobre mqs.ini, consulte "Archivo de configuración de IBM MQ, mqs.ini" en la página 96.

Localice la stanza `QueueManager` para el gestor de colas en el archivo. El directorio de datos del gestor de datos es el valor de la clave llamada `DataPath`. Para obtener más información sobre la stanza `QueueManager`, consulte "Stanza QueueManager del archivo mqs.ini" en la página 107.

d) Cree una copia de seguridad del directorio de datos de gestor de colas. En Linux, puede hacerlo utilizando el mandato **tar**. Por ejemplo, para hacer una copia de seguridad del directorio de datos para un gestor de colas, puede utilizar el mandato siguiente. Observe el último parámetro del mandato, que es un único carácter de punto (.):

```
tar -cvzf qm-data.tar.gz -C queue_manager_data_dir .
```

e) Determine la ubicación del directorio de registro del gestor de colas visualizando el archivo de configuración del gestor de colas de IBM MQ `qm.ini`. Este archivo se encuentra en el directorio de datos del gestor de colas. Para obtener más información sobre el archivo, consulte "Archivos de configuración de gestores de colas, qm.ini" en la página 109.

El directorio de registro del gestor de colas se define como el valor de la clave `LogPath` en la stanza Log. Para obtener información sobre la stanza, consulte "Stanza de registro del archivo qm.ini" en la página 145.

f) Cree una copia de seguridad del directorio de registro del gestor de colas. En Linux, puede hacerlo utilizando el mandato **tar**. Por ejemplo, para hacer una copia de seguridad del directorio de registro para un gestor de colas, puede utilizar el mandato siguiente. Observe el último parámetro del mandato, que es un único carácter de punto (.):

```
tar -cvzf qm-log.tar.gz -C queue_manager_log_dir .
```

g) Suprima el gestor de colas RDQM existente.

2. Restaure el gestor de colas con un sistema de archivos del tamaño necesario:

a) Cree un nuevo gestor de colas RDQM con el mismo nombre que el gestor de colas del que ha hecho una copia de seguridad. Asegúrese de que el sistema de archivos asignado para el gestor de colas RDQM por **crtmqm** es el tamaño que requiere, y es lo suficientemente grande para contener los datos, registros primarios y registros secundarios para el gestor de colas existente, además de algo de espacio adicional para la futura expansión. Para obtener información sobre cómo crear un gestor de colas RDQM, consulte "Creación de un RDQM de HA" en la página 605.

b) Determine el nodo RDQM primario para el gestor de colas. Para obtener más información sobre cómo determinar el nodo primario, consulte rdqmstatus (mostrar estado RDQM).

c) En el nodo RDQM primario, si se ha iniciado el gestor de colas RDQM, deténgalo utilizando el mandato **endmqm -w** o **endmqm -i** .

d) En el nodo RDQM primario, determine la nueva ubicación de los datos y los directorios de registro para el gestor de colas RDQM (utilice los métodos descritos en los pasos 1c y 1e).

e) En el nodo RDQM primario, suprima el contenido de los directorios de datos y registros del gestor de colas RDQM, pero no los propios directorios.

f) En el nodo RDQM primario, restaure la copia de seguridad del directorio de datos del gestor de colas en el directorio de datos vacío para el nuevo gestor de colas RDQM, asegurándose de que se conserven los permisos y la propiedad de archivos. Si la copia de seguridad se ha creado utilizando el mandato **tar** de ejemplo en el paso 1d, el usuario root puede utilizar el mandato siguiente para restaurarla:

```
tar -xvzpf qm-data.tar.gz -C queue_manager_data_dir
```

g) En el nodo RDQM primario, restaura la copia de seguridad del directorio de registro del gestor de colas en el directorio de registro vacío para el nuevo gestor de colas RDQM, asegurándose de que se conserven los permisos y la propiedad de archivos. Si la copia de seguridad se ha creado utilizando el mandato **tar** de ejemplo en el paso 1f, el usuario root puede utilizar el mandato siguiente para restaurarla:

```
tar -xvzpf qm-log.tar.gz -C queue_manager_log_dir
```

h) En el nodo RDQM primario, edite el archivo de configuración del gestor de colas restaurado, `qm.ini`, en el directorio de datos del nuevo gestor de colas RDQM. Actualice el valor de la clave `LogPath` en la stanza `Log` para especificar el directorio de registro para el nuevo gestor de colas RDQM que ha determinado en el paso 2d. Revise otras vías de acceso de archivo que están definidas en el archivo de configuración y actualícelas, si es necesario. Por ejemplo, es posible que tenga que actualizar las vías de acceso siguientes:

- La vía de acceso de los archivos de registro de errores generados por los servicios de mensajes de diagnóstico.
- La vía de acceso para las salidas necesarias para el gestor de colas.
- La vía de acceso para los archivos de carga conmutada, si el gestor de colas es un coordinador de transacciones XA.

i) Verifique que el gestor de colas se muestra mediante el mandato **dspmq** y que su estado se notifica como finalizado. El ejemplo siguiente muestra la salida de ejemplo para un gestor de colas RDQM HA.

```
$ dspmq -o status -o ha
QMNAME(QM1) STATUS(Ended normally) HA(Replicated)
```

j) Verifique que los datos del gestor de colas restaurado se han duplicado en los nodos RDQM secundarios utilizando el mandato **rdqmstatus** para mostrar el estado del gestor de colas. El estado de HA se debe notificar como `Normal` en cada nodo. El ejemplo siguiente muestra la salida de ejemplo para un gestor de colas RDQM HA.

```
$ rdqmstatus -m QM1
Node: mqhavm10-adm
Queue manager status:         Ended normally
Queue manager file system:    50MB used, 0.2GB
allocated [42%]
HA role:                      Primary
HA status:                    Normal
HA control:                   Disabled
HA current location:          This node
HA preferred location:        This node
HA floating IP interface:     None
HA floating IP address:       None
Node:                         mqhavm11-adm
HA status:                    Normal
Node:                         mqhavm12-adm
HA status:                    Normal
```

k) Inicie el gestor de colas en el nodo RDQM primario.

l) Realice una migración tras error gestionada del gestor de colas a cada nodo RDQM para garantizar que se ha establecido correctamente la configuración necesaria, consulte “Configuración de la ubicación preferida de un RDQM” en la página 614.

Almacenamiento del estado de aplicación persistente
Puede almacenar información de estado persistente relacionada con aplicaciones junto con los datos de gestor de colas.

Cada gestor de colas de IBM MQ tiene un sistema de archivos dedicado para su estado persistente, que incluye tanto sus datos de cola como el registro de recuperación. En una configuración RDQM un volumen lógico que se replica entre los sistemas Linux (nodos) respalda el sistema de archivos. El sistema de archivos incluye un directorio `userdata` que puede utilizar para almacenar información de estado persistente para las aplicaciones. Por lo tanto, cuando un gestor de colas de datos duplicados se traslada para ejecutarse en otro nodo de la configuración RDQM, tiene disponible el contexto de aplicación, así como el contexto del gestor de colas. Consulte Contenido de directorio en sistemas Unix y Linux.

Si elige almacenar el estado de aplicación en el directorio `userdata`, debe tener en cuenta que los datos escritos en esta ubicación pueden consumir el espacio de disco disponible asignado al gestor de colas. Debe asegurarse de que haya suficiente espacio de disco disponible para que el gestor de colas escriba datos de cola, registros y otra información de estado persistente.

El directorio `userdata` tiene la propiedad de usuario y grupo mqm y lo puede leer todo el mundo para que los usuarios puedan acceder al mismo sin necesidad de pertenecer al grupo de administradores de IBM MQ (es decir, mqm). No puede modificar los permisos del directorio `userdata`, pero puede crear contenido en él con la propiedad y los permisos necesarios.

Durante la migración tras error del gestor de colas RDQM, finaliza el gestor de colas y se desmonta su sistema de archivos en el nodo RDQM actual. Después, se monta el sistema de archivos y se reinicia el gestor de colas en otro nodo de la configuración RDQM. No se puede desmontar un sistema de archivos si un proceso tiene un manejador abierto para uno de sus archivos. Para asegurarse de que se puede completar una migración tras error del gestor de colas, si no se puede desmontar el sistema de archivos del gestor de colas, se envía una señal SIGTERM a los procesos que tienen un manejador de archivos abierto, seguida de una señal SIGKILL si no se han liberado los manejadores abiertos. Las aplicaciones deben estar diseñadas para responder correctamente a SIGTERM. Si las aplicaciones o los procesos se configuran como un servicio de gestor de colas, durante una migración tras error gestionada, se pueden finalizar durante la conclusión del gestor de colas antes de que se desmonte el sistema de archivos. Si una aplicación o un proceso no está configurado como un servicio de gestor de colas o se produce una migración tras error no gestionada, como por ejemplo una pérdida de quórum, es probable que se envíen señales para liberar el sistema de archivos.

Linux *Configuración de la ubicación preferida de un RDQM*

La ubicación preferida de un gestor de colas de datos replicados (RDQM) identifica el nodo donde RDQM tiene que ejecutar si dicho nodo está disponible.

Acerca de esta tarea

La ubicación preferida es el nombre del nodo en el que Pacemaker tiene que ejecutar el gestor de colas cuando el grupo HA se encuentra en un estado normal (todos los nodos y conexiones disponibles). La ubicación preferida se inicializa al nombre del nodo primario cuando se crea el gestor de colas. Los mandatos para definir la ubicación preferida se pueden ejecutar en cualquiera de los tres nodos. Hay que ser un usuario que pertenezca a los grupos `mqm` y `haclient`.

Procedimiento

- Para asignar el nodo local o especificado como ubicación preferida del gestor de colas con nombre, ejecute el siguiente mandato:

  ```
  rdqmadm -p -m qmname [ -n nodename[,nodename  ]
  ```

  donde *nombreqm* es el nombre del RDQM cuya ubicación preferida se está especificando y *nombrenodo* es opcionalmente el nombre del nodo preferido.

  Si el grupo HA se encuentra en un estado normal y la ubicación preferida no es el nodo primario actual, el gestor de colas se para y se reinicia en la nueva ubicación preferida. Se puede especificar una lista separada por comas de dos nombres de nodo para asignar una ubicación preferida alternativa.
- Para borrar la ubicación preferida a fin de que el gestor de colas no vuelva automáticamente a un nodo cuando se restaure, ejecute el mandato siguiente:

  ```
  rdqmadm -p -m qmname -d
  ```

Referencia relacionada

rdqmadm (administrar un clúster de gestores de colas de datos replicados)

Linux *Creación y borrado de una dirección IP flotante*

Una dirección IP flotante permite que un cliente utilice la misma dirección IP para un gestor de colas de datos replicados (RDQM) independientemente del nodo del grupo HA en que esté ejecutando. (El uso de una dirección IP flotante es opcional).

Acerca de esta tarea

Se puede crear o borrar una dirección IP flotante con el mandato **rdqmint**. La dirección flotante vincula con una interfaz física con nombre en el nodo primario del RDQM. Si el RDQM migra tras error y pasa a ser primario un nodo diferente, la dirección IP flotante está vinculada a una interfaz del mismo nombre en el nuevo primario. Las interfaces físicas de los tres nodos tienen que pertenecer a la misma subred que la dirección IP flotante. El diagrama siguiente ilustra el uso de una dirección IP flotante.

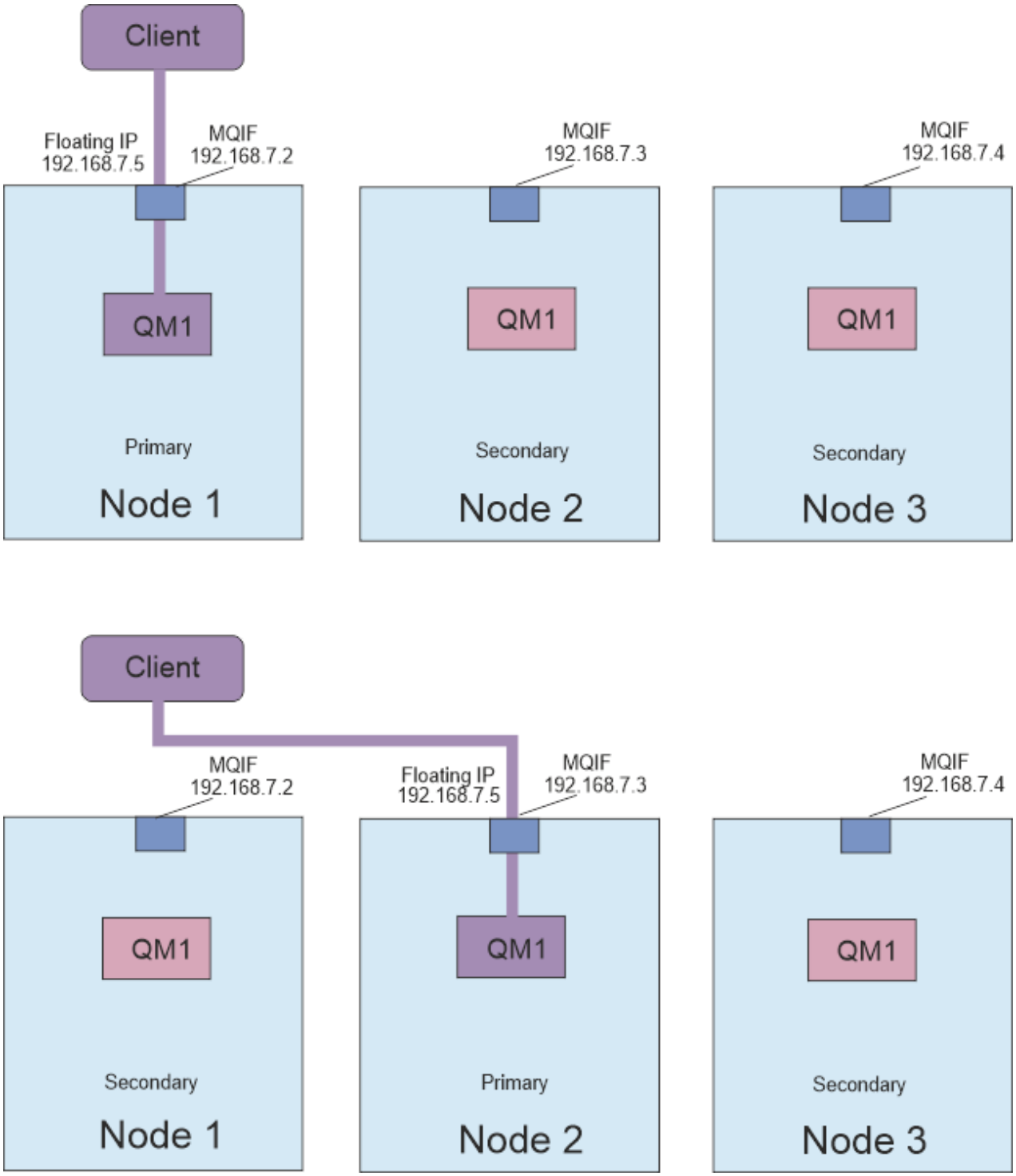


Figura 80. Dirección IP flotante

Hay que ser un usuario en los grupos `mqm` y `haclient` para ejecutar el mandato **rdqmint**. Se puede crear o borrar la dirección IP flotante en el nodo primario del RDQM o en cualquiera de los nodos secundarios.

Nota: No puede utilizar la misma dirección IP flotante para varios RDQM, la dirección IP flotante para cada RDQM debe ser única.

Procedimiento

- Para crear la dirección IP flotante de un RDQM, ejecute el mandato siguiente:

  ```
  rdqmint -m qmname -a  -f ipv4address -l interfacename
  ```

  donde:

  nombregc
  Es el nombre del RDQM cuya dirección IP flotante se está creando.

  ipv4address
  Es la dirección IP flotante en formato ipv4.

  La dirección IP flotante debe ser una dirección IPv4 válida que no esté ya definida en ningún nodo HA, y debe pertenecer a la misma subred que las direcciones IP estáticas definidas para la interfaz local.

  nombreinterfaz
  Es el nombre de la interfaz física en el nodo primario con la que hay que vincularse.

  Por ejemplo:

  ```
  rdqmint -m QM1 -a -f 192.168.7.5 -l MQIF
  ```

- Para borrar una dirección IP flotante existente, especifique el mandato siguiente:

  ```
  rdqmint -m qmname -d
  ```

Referencia relacionada
rdqmint (añadir o suprimir dirección IP flotante para RDQM)

Linux *Inicio, detención y visualización del estado de un RDQM de HA*

Se usan variantes de mandatos de control de IBM MQ para iniciar, parar y ver el estado actual de un gestor de colas de datos replicados (RDQM).

Acerca de esta tarea

Hay que ejecutar los mandatos que inician, paran y visualizan el estado actual de un gestor de colas de datos replicados (RDQM) con un usuario que pertenezca a los grupos `mqm` y `haclient`.

Debe ejecutar los mandatos para iniciar y detener un gestor de colas en el nodo primario para ese gestor de colas.

Procedimiento

- Para iniciar un RDQM, ejecute el mandato siguiente en el nodo primario del RDQM:

  ```
  strmqm qmname
  ```

  donde *nombreGC* es el nombre del RDQM que se quiere iniciar.

  El RDQM se inicia y Pacemaker empieza a gestionar el RDQM. Hay que especificar la opción `-ns` con `strmqm` si se desea especificar cualquier otra opción `strmqm`.

- Para parar un RDQM, ejecute el mandato siguiente en el nodo primario del RDQM:

  ```
  endmqm qmname
  ```

  donde *nombreGC* es el nombre del RDQM que se desea parar.

Pacemaker deja de gestionar el RDQM y este se termina. Todos los demás parámetros **endmqm** se pueden utilizar cuando se para un RDQM.

- Para ver el estado un RDQM, ejecute el mandato siguiente:

```
dspmq
```

La información de estado que aparece en la salida depende de si se ejecuta el mandato en el nodo primario o secundario del RDQM. Si se ejecuta en el nodo primario, se mostrará uno de los mensajes de estado normal devueltos por **dspmq**. Si se ejecuta el mandato en un nodo secundario, se muestra el estado `running elsewhere` (ejecutando en otra parte). Por ejemplo, si se ejecuta **dspmq** en el nodo RDQM7, podría devolverse la siguiente información:

```
QMNAME(RDQM8)                           STATUS(Running elsewhere)
QMNAME(RDQM9)                           STATUS(Running elsewhere)
QMNAME(RDQM7)                           STATUS(Running)
```

Si el nodo primario no está disponible, o si se ejecuta **dspmq** con un usuario que sea `root` o un miembro del grupo `haclient`, se notifica el estado `Unavailable` (no disponible). Por ejemplo:

```
QMNAME(RDQM8)               STATUS(Unavailable)
QMNAME(RDQM9)               STATUS(Unavailable)
QMNAME(RDQM7)               STATUS(Unavailable)
```

Puede especificar el mandato **dspmq -o ha** (o **dspmq -o HA**) para ver una lista de los gestores de colas conocidos por un nodo, y si son o no RDQM, por ejemplo:

```
dspmq -o ha

QMNAME(RDQM8)                                           HA(Replicated)
QMNAME(RDQM9)                                           HA(Replicated)
QMNAME(RDQM7)                                           HA(Replicated)
QMNAME(QM7)                                             HA()
```

Referencia relacionada

dspmq (visualizar gestores de colas)
endmqm (finalizar gestor de colas)
strmqm (iniciar gestor de colas)

Acciones de recurso fallido

Las acciones de recurso fallido surgen cuando el componente Pacemaker de una configuración de alta disponibilidad en RDQM encuentra algún problema con un recurso en uno de los nodos de un grupo HA.

La solución HA de RDQM utiliza Pacemaker para supervisar y gestionar los recursos (consulte “Alta disponibilidad en RDQM” en la página 595). Si Pacemaker encuentra un error al realizar una operación en un recurso de un nodo, registra esta información utilizando una acción de recurso fallido. Algunas acciones de recurso fallido impiden que el recurso se ejecute y deben borrarse antes de que Pacemaker pueda reiniciar el recurso.

Puede utilizar el mandato **rdqmstatus -m** para ver si hay alguna acción de recurso fallido que impida que un gestor de colas se inicie en uno o más nodos.

A continuación, puede utilizar el mandato **rdqmstatus -m** *nombreGC* **-a** para ver los detalles de las acciones de recurso fallido asociadas con un gestor de colas. Seguidamente, utilice el mandato **rdqmclean** para borrar estas acciones de recurso fallido y liberar así los recursos restringidos. En cualquier caso, primero debe solucionar los problemas que han causado la acción de recurso fallido.

Pacemaker controla los siguientes recursos en una configuración de HA de RDQM y pueden ser el sujeto de acciones de recurso fallido:

- Gestor de colas
- IP flotante
- Control de RDQM

- Sistema de archivos
- Réplica de DR (DRBD)
- Réplica de HA (DRBD)

Cada tipo de recurso puede estar sujeto a los siguientes tipos de anomalía:

Leve
Las anomalías leves son transitorias y Pacemaker sigue intentando recuperar el recurso hasta que se excede el tiempo de espera o se detiene de alguna otra manera.

Grave
Un error grave requiere intervención administrativa. Los errores graves bloquean la ejecución del recurso en un nodo en particular.

Muy grave
Un error muy grave requiere intervención administrativa. Los errores muy graves bloquean la ejecución del recurso en cualquier nodo.

Consulte "Visualización del estado de un RDQM y de un grupo HA" en la página 618 para ver ejemplos de estados que incluyen acciones de cola de recursos fallidos.

Puede utilizar el mandato **rdqmclean** para borrar todas las acciones de recurso fallido asociadas con un gestor de colas concreto o bien todas las acciones de recurso fallido de la configuración de HA de RDQM.

Nota: Algunas acciones de recurso fallido no provocan el bloqueo del gestor de colas en un nodo. Por ejemplo, después de una finalización inesperada del gestor de colas, Pacemaker intenta reiniciar el gestor de colas en el nodo donde no se estaba ejecutando. Si el inicio es satisfactorio, no se bloquea la ejecución del gestor de colas en el nodo. En este caso, la única manera en la que el usuario podría darse cuenta de la acción de recurso fallido sería ejecutando el mandato **rdqmstatus -m** *nombreGC* **-a**.

Tareas relacionadas
"Visualización del estado de un RDQM y de un grupo HA" en la página 618
Se puede ver el estado de un grupo HA y de gestores de colas de datos replicados (RDQM) individuales.

Referencia relacionada
rdqmclean
rdqmstatus

Linux *Visualización del estado de un RDQM y de un grupo HA*

Se puede ver el estado de un grupo HA y de gestores de colas de datos replicados (RDQM) individuales.

Acerca de esta tarea

El mandato **rdqmstatus** se usa para ver el estado de un RDQM individual y de un grupo HA en su conjunto.

El estado de resumen de un nodo también muestra información sobre el módulo de kernel de DRBD en el que se basa RDQM. Cuando se actualiza RDQM, es importante asegurarse de que se haya instalado la versión correcta del módulo de kernel de DRBD para la versión del kernel RHEL que se ejecuta en el sistema. El estado muestra la versión del kernel de sistema operativo, la versión del kernel para la que se ha creado el módulo DRBD, la versión de DRBD y el estado cargado del módulo de kernel de DRBD.

Debe ser un usuario de los grupos `mqm` y `haclient` para ejecutar el mandato **rdqmstatus**. Se puede ejecutar el mandato en cualquiera de los tres nodos.

Procedimiento

- Para ver el estado de resumen de un nodo y los RDQM de forman parte de la configuración HA:

```
rdqmstatus
```

Se visualiza la identidad del nodo que ha ejecutado el mandato en, los detalles de kernel y DRBD para dicho nodo, y el estado de los RDQM en la configuración de HA, por ejemplo:

```
Node:                                 mqhavm07.exampleco.com
OS kernel version:                    5.14.0-362.18.1
DRBD OS kernel version:               5.14.0-362.18.1
DRBD version:                         9.2.7
DRBD kernel module status:            Loaded

Queue manager name:                   RDQM8
Queue manager status:                 Running elsewhere
HA current location:                  mqhavm08.exampleco.com
HA preferred location:                mqhavm08.exampleco.com
HA blocked location:                  None

Queue manager name:                   RDQM9
Queue manager status:                 Running elsewhere
HA current location:                  mqhavm09.exampleco.com
HA preferred location:                mqhavm09.exampleco.com
HA blocked location:                  None

Queue manager name:                   RDQM7
Queue manager status:                 Running
HA current location:                  This node
HA preferred location:                This node
HA blocked location:                  None
```

El estado del módulo de kernel de DRBD tiene uno de los siguientes valores:

Cargado
Indica que se ha cargado el módulo DRBD.

Cargado parcialmente
Puede producirse cuando se ha cargado el módulo DRBD, pero no funciona correctamente debido a una discrepancia.

No cargado
No se ha cargado el módulo DRBD. Esto se puede visualizar en una configuración recién instalada, cuando todavía no se ha creado ningún gestor de colas RDQM.

No instalado
Indica que el módulo DRBD no está instalado. o que IBM MQ no ha podido determinar la versión de kernel del sistema operativo del módulo DRBD.

Versión instalada anteriormente todavía cargada
Este estado puede producirse si se instala un nuevo módulo DRBD mientras se ejecuta el módulo DRBD existente (es decir, se está ejecutando un gestor de colas RDQM). El módulo recién instalado se notifica en el estado, pero no es el módulo que se está ejecutando realmente.

- Para ver el estado de los tres nodos del grupo HA, escriba el mandato siguiente:

```
rdqmstatus -n
```

Se notifica el estado en línea o fuera de línea de cada nodo. Por ejemplo:

```
Node mqha04(mqhavm04.example.com) is online
Node mqha05(mqhavm05.example.com) is offline
Node mqha06(mqhavm06.example.com) is online
```

- Para ver el estado de un gestor de colas determinado en todos los nodos del grupo HA, ejecute el mandato siguiente:

```
rdqmstatus -m qmname
```

donde *nombreGC* es el nombre del RDQM cuyo estado se desea visualizar. Se muestra el estado del RDQM del nodo actual, seguido de un resumen del estado de los otros dos nodos desde la perspectiva del nodo actual.

- Para ver el estado de un gestor de colas determinado en todos los nodos del grupo HA, incluyendo los detalles de posibles acciones de recurso fallido, ejecute el mandato siguiente:

```
rdqmstatus -m qmname -a
```

donde *nombreGC* es el nombre del RDQM cuyo estado se desea visualizar. Se muestra el estado del RDQM del nodo actual, seguido de un resumen del estado de los otros dos nodos desde la perspectiva del nodo actual. Esto va acompañado de los detalles de cualquier posible acción de recurso fallido asociada con el RDQM.

- La tabla siguiente resume la información sobre el nodo actual que puede devolver el mandato `rdqmstatus -m qmname` para un RDQM.

Tabla 33. Estado de nodo actual

Atributo de estado	**Valores posibles**	**Cuándo se muestra**
Nombre de nodo	*nombrenodo*	Siempre se muestra
Estado del gestor de colas	En ejecución Ejecución en otro sitio Finalizado No disponible	Siempre se muestra
CPU	*n.nn*%	Solo se muestra cuando el nodo actual tiene el un rol principal (es decir, el RDQM ejecuta en este nodo)
Memoria	*nnn*MB usadas, *y.y*GB asignadas	Solo se muestra cuando el nodo actual tiene el un rol principal (es decir, el RDQM ejecuta en este nodo)
Sistema de archivos del gestor de colas	*nnn*MB usadas, *y.y*GB asignadas [*z*%]	Solo se muestra cuando el nodo actual tiene el un rol principal (es decir, el RDQM ejecuta en este nodo)
Rol de HA	Primario Secundario Desconocido	Siempre se muestra
Estado de HA	Todos los nodos en espera Este nodo en espera Nodos remotos en espera Mixto *estado de los nodos remotos*	Todos los nodos en espera Nodo actual en espera Ambos nodos remotos en espera Distintos estados por cada nodo remoto (consulte la tabla siguiente para ver los estados individuales) Mismo estado en ambos nodos remotos (consulte la tabla siguiente para obtener todos los valores)
Control de HA	Habilitada Inhabilitado Desconocido	Siempre se muestra. Indica si RDQM está bajo el control de Pacemaker
Ubicación de HA preferida	Ninguna Este nodo Desconocido *nombrenodo*	Siempre se muestra

Tabla 33. Estado de nodo actual (continuación)		
Atributo de estado	**Valores posibles**	**Cuándo se muestra**
Ubicación bloqueada de HA	Ninguna - El gestor de colas no está bloqueado y puede ejecutarse en cualquier nodo Este nodo - El gestor de colas está bloqueado y no puede ejecutarse en el nodo actual debido a una o más acciones de recurso fallido *nombrenodo* - El gestor de colas está bloqueado y no puede ejecutarse en el nodo *nombrenodo* debido a una o más acciones de recurso fallido *nombrenodo1*, *nombrenodo2* - El gestor de colas está bloqueado y no puede ejecutarse en *nombrenodo1* y *nombrenodo2* debido a una o más acciones de recurso fallido Todos los nodos - El gestor de colas está bloqueado y no puede ejecutarse en ningún nodo debido a una o más acciones de recurso fallido	Siempre se muestra
Interfaz de IP flotante de HA	*nombre_interfaz*	Siempre se muestra
Dirección IP flotante de HA	*IPV4_address*	Siempre se muestra

La tabla siguiente resume la información que devuelve el mandato `rdqmstatus -m` *qmname* para los otros nodos del grupo HA.

Tabla 34. Estado de otros nodos		
Atributo de estado	**Valores posibles**	**Cuándo se muestra**
Nombre de nodo	*nodename*	Siempre se muestra
Estado de HA	Normal Sincronización en curso Remoto no disponible Incoherente En pausa Nodo remoto en espera Desconocido	Nodos sincronizados entre sí Sincronizando con el nodo remoto No se puede comunicar con el nodo remoto Sin sincronizar con el nodo remoto y no sincronizando Réplica en pausa Nodo remoto en espera
Sincronización HA en curso	*n.n*%	Se visualiza cuando la sincronización está en curso y el mandato se ejecuta como `root`
Hora de sincronización HA estimada	*aaaa-mm-dd hh:mm:ss.nnn*	Se muestra cuando la sincronización está en curso

Tabla 34. Estado de otros nodos (continuación)		
Atributo de estado	**Valores posibles**	**Cuándo se muestra**
Datos HA sin sincronizar	*n*KB	Se muestra cuando el nodo remoto no está disponible o no es coherente
Última sincronización de HA	*aaaa-mm-dd hh:mm:ss.nnn*	Se muestra cuando los datos de HA están sin sincronizar (después de la sincronización inicial). Proporciona la hora y la fecha cuando los datos se han sincronizado por última vez.

Ejemplo

Ejemplo de estado normal en el nodo primario:

```
Node:                                  mqhavm07.exampleco.com
Queue manager status:                   Running
CPU:                                    0.00
Memory:                                 123MB
Queue manager file system:              606MB used, 1.0GB allocated [60%]
HA role:                                Primary
HA status:                              Normal
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA preferred location:                  This node
HA blocked location:                    None
HA floating IP interface:               eth4
HA floating IP address:                 192.0.2.4


Node:                                   mqhavm08.exampleco.com
HA status:                              Normal


Node:                                   mqhavm09.exampleco.com
HA status:                              Normal
```

Ejemplo de estado normal en un nodo secundario:

```
Node:                                   mqhavm08.exampleco.com
Queue manager status:                   Running elsewhere
HA role:                                Secondary
HA status:                              Normal
HA control:                             Enabled
HA current location:                    mqhavm07.exampleco.com
HA preferred location:                  mqhavm07.exampleco.com
HA blocked location:                    None
HA floating IP interface:               eth4
HA floating IP address:                 192.0.2.4

Node:                                   mqhavm07.exampleco.com
HA status:                              Normal

Node:                                   mqhavm09.exampleco.com
HA status:                              Normal
```

Ejemplo de estado en el nodo primario cuando la sincronización está en curso:

```
Node:                                   mqhavm07.exampleco.com
Queue manager status:                   Running
CPU:                                    0.53
Memory:                                 124MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
HA role:                                Primary
HA status:                              Synchronization in progress
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA blocked location:                    None
```

```
HA floating IP interface:               eth4
HA floating IP address:                 192.0.2.4

Node:                                   mqhavm08.exampleco.com
HA status:                              Synchronization in progress
HA synchronization progress:            11.0%
HA estimated time to completion:        2017-09-06 14:55:05

Node:                                   mqhavm09.exampleco.com
HA status:                              Synchronization in progress
HA synchronization progress:            11.0%
HA estimated time to completion:        2017-09-06 14:55:06
```

Ejemplo de estado en el nodo primario cuando la sincronización se ha perdido:

```
Node:                                   mqhavm07.exampleco.com
Queue manager status:                   Running
CPU:                                    0.53
Memory:                                 124MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
HA role:                                Primary
HA status:                              Mixed
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA blocked location:                    None
HA floating IP interface:               eth4
HA floating IP address:                 192.0.2.4

Node:                                   mqhavm08.exampleco.com
HA status:                              Normal

Node:                                   mqhavm09.exampleco.com
HA status:                              Inconsistent
HA out of sync data:                    15932KB
HA last in sync:                        2017-09-06 14:55:06
```

Ejemplo de un nodo primario que muestra varios estados:

```
Node:                                   mqhavm07.exampleco.com
Queue manager status:                   Running
CPU:                                    0.02
Memory:                                 124MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
HA role:                                Primary
HA status:                              Mixed
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA blocked location:                    None
HA floating IP interface:               eth4
HA floating IP address:                 192.0.2.4

Node:                                   mqhavm08.exampleco.com
HA status:                              Normal

Node:                                   mqhavm09.exampleco.com
HA status:                              Inconsistent
```

Ejemplo de un nodo primario que muestra acciones de recurso fallido:

```
Node:                                   mqhavm07.exampleco.com
Queue manager status:                   Running
CPU:                                    0.00%
Memory:                                 123MB
Queue manager file system:              606MB used, 1.0GB allocated [60%]
HA role:                                Primary
HA status:                              Normal
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  mqhavm08.exampleco.com
HA blocked location:                    mqhavm08.exampleco.com
HA floating IP interface:               eth4
HA floating IP address:                 192.0.2.4

Node:                                   mqhavm08.exampleco.com
HA status:                              Normal
```

```
Node:                                   mqhavm09.exampleco.com
HA status:                              Normal

Failed resource action:                 Start
Resource type:                          Filesystem
Failure node:                           mqhavm08.exampleco.com
Failure time:                           2017-09-06 12:00:00
Failure reason:                         Couldn't find directory  [/var/mqm/vols/qmname] to use
as a mount point
Blocked location:                       mqhavm08.exampleco.com
```

Este estado muestra que Pacemaker no ha podido iniciar el sistema de archivos en el nodo mqhavm08.exampleco.com a las 12:00:00. Esta acción de recurso fallido significa que la ejecución del gestor de colas está bloqueada en mqhavm08.exampleco.com. Una vez que se haya resuelto el problema subyacente causante de la acción de recurso fallido, ejecute el mandato **rdqmclean** para borrar la acción fallida de manera que Pacemaker pueda reintentar la acción (si es necesario).

Ejemplo de un estado de resumen que muestra una discrepancia entre la versión del kernel del sistema operativo (RHEL 9.3) y el módulo de kernel DRBD (destinado a RHEL 9.2). Aunque el estado indica que se ha cargado el módulo de kernel de DRBD y se está ejecutando el gestor de colas, en esta situación debe actualizar el módulo de kernel de DRBD con la versión destinada al kernel del sistema operativo en ejecución.

```
Node:                                   mqhavm07.exampleco.com
OS kernel version:                      5.14.0-362.18.1
DRBD OS kernel version:                 5.14.0-284.11.1
DRBD version:                           9.2.7+ptf.14
DRBD kernel module status:              Loaded

Queue manager name:                     RDQM7
Queue manager status:                   Running
HA current location:                    This node
HA preferred location:                  This node
HA blocked location:                    None
```

Ejemplo de un estado de resumen que muestra una discrepancia entre la versión de kernel del sistema operativo (RHEL 8.10) y el módulo de kernel DRBD (destinado a RHEL 8.8). En este ejemplo, la discrepancia de versiones es más grave y el módulo de kernel de DRBD no se puede cargar correctamente. Como resultado, el gestor de colas no se puede iniciar en su nodo preferido y su estado de HA en Unknown. Para resolver esta anomalía, el módulo de kernel de DRBD debe actualizarse con la versión de destino para el kernel del sistema operativo en ejecución.

```
Node:                                   mqhavm57.exampleco.com
OS kernel version:                      4.18.0-553
DRBD OS kernel version:                 4.18.0-477
DRBD version:                           9.2.7+ptf.14
DRBD kernel module status:              Partially loaded

Queue manager name:                     QM2
Queue manager status:                   Running elsewhere
HA status:                              Unknown
HA current location:                    mqhavm58.exampleco.com
HA preferred location:                  This node
HA blocked location:                    All nodes
```

Referencia relacionada

Linux rdqmstatus

Modificación de las direcciones IP en configuraciones de alta disponibilidad

Si cambia las direcciones IP de cualquiera de las interfaces en una configuración de alta disponibilidad, la operación de alta disponibilidad ya no está disponible y el gestor de colas no se ejecutará en el nodo donde se han modificado las direcciones.

Puede especificar hasta tres direcciones IP para la operación de alta disponibilidad en el archivo `rdqm.ini`. Si ya ha cambiado las direcciones del supervisor de Pacemaker, debe restaurarlas temporalmente en sus valores originales antes de seguir el procedimiento. De lo contrario, no es posible suprimir el gestor de colas RDQM de alta disponibilidad.

1. Elimine la configuración de alta disponibilidad en cada nodo. Puede eliminar la alta disponibilidad haciendo una copia de seguridad de los gestores de colas y, a continuación, suprimirlos, consulte "Copia de seguridad y restauración de datos de gestor de colas de IBM MQ" en la página 706 y "Supresión de un RDQM de HA" en la página 607 y, a continuación, eliminando el propio grupo HA, consulte "Supresión del clúster de Pacemaker (grupo HA)" en la página 605.
2. Vuelva a crear la configuración de alta disponibilidad con las nuevas direcciones IP, consulte "Definición del clúster de Pacemaker (grupo HA)" en la página 601.
3. Vuelva a crear los gestores de colas de alta disponibilidad y restaure la copia de seguridad, consulte "Creación de un RDQM de HA" en la página 605 y "Copia de seguridad y restauración de datos de gestor de colas de IBM MQ" en la página 706.

Linux *Sustitución de un nodo que ha fallado en una configuración de disponibilidad*

Si uno de los nodos del grupo HA falla, puede sustituirse.

Acerca de esta tarea

Los pasos a seguir para sustituir un nodo dependen del escenario:

- Si va a sustituir el nodo que ha fallado por un nodo con una configuración idéntica, dicho nodo se puede sustituir sin interrumpir el grupo HA.
- Si el nodo nuevo tiene una configuración diferente, hay que eliminar el grupo HA y volver a crearlo. En primer lugar, puede hacer una copia de seguridad de los gestores de colas del nodo en el que se ejecutan y luego restaurarlos después de haber reconstruido el grupo HA.

Procedimiento

- Si el nodo de sustitución está configurado para parecerse al nodo fallido (mismo nombre de host, mismas direcciones IP, etc.), siga los pasos siguientes en el nodo nuevo:
  a) Cree un archivo `rdqm.ini` que coincida con los archivos de los demás nodos y a continuación ejecute el mandato `rdqmadm -c` (consulte "Definición del clúster de Pacemaker (grupo HA)" en la página 601).
  b) Ejecute el mandato `crtmqm -sxs` *qmanager* para volver a crear cada gestor de colas de datos replicados (consulte "Creación de un RDQM de HA" en la página 605).
- Si el nodo de sustitución tiene una configuración diferente de la del nodo fallido:
  a) Si es necesario, vuelva a realizar la copia de seguridad de los gestores de colas (consulte "Copia de seguridad y restauración de datos de gestor de colas de IBM MQ" en la página 706).
  b) Borre los gestores de colas de datos replicados en los otros nodos del grupo HA con el mandato **dltmqm** (consulte "Supresión de un RDQM de HA" en la página 607).
  c) Desconfigure el clúster Pacemaker utilizando el mandato **rdqmadm -u** (consulte "Supresión del clúster de Pacemaker (grupo HA)" en la página 605).
  d) Vuelva a configurar el clúster de Pacemaker, incluyendo la información del nodo nuevo, con el mandato **rdqmadm -c** (consulte "Definición del clúster de Pacemaker (grupo HA)" en la página 601).
  e) Si es necesario (es decir, si no tiene acceso SSH a los otros nodos), ejecute el mandato `crtmqm -sxs` *qmanager* para volver a crear cada gestor de colas de datos replicados en los otros nodos (consulte "Creación de un RDQM de HA" en la página 605).
  f) Ejecute el mandato `crtmqm -sx` *qmanager*para crear los gestores de colas en el nodo de sustitución.
  g) Si es necesario, restaure los datos y la configuración de los gestores de colas (consulte "Copia de seguridad y restauración de datos de gestor de colas de IBM MQ" en la página 706).

MQ Adv. Linux Recuperación tras desastre de RDQM

RDQM (gestor de cola de datos replicados) está disponible en un subconjunto de plataformas Linux y puede proporcionar una solución de recuperación tras desastre.

Para ver la información detallada, consulte Software Product Compatibility Reports.

Puede crear una instancia primaria de un gestor de colas de recuperación tras desastre en ejecución en un servidor, y una instancia secundaria del gestor de colas en otro servidor que actúe como nodo de recuperación. Los datos se replican entre las instancias del gestor de colas. Si pierde el gestor de colas primario, puede convertir manualmente la instancia secundaria en la instancia primaria e iniciar el gestor de colas, y a continuación reanudar el trabajo desde el mismo punto. No puede iniciar un gestor de colas mientras este esté en el rol secundario. La réplica de los datos entre los dos nodos la gestiona DRBD.

Puede elegir entre réplica síncrona y asíncrona de datos entre los gestores de cola primario y secundario. Si selecciona la opción asíncrona, las operaciones como PUT o GET de IBM MQ se completan y devuelven a la aplicación antes de que el suceso se replique en el gestor de colas secundario. La réplica asíncrona significa que, tras una situación de recuperación, es posible que se pierdan algunos datos de mensajes. Sin embargo, el gestor de colas secundario estará en un estado coherente, y podrá empezar a ejecutarse inmediatamente, aunque se inicie en un parte ligeramente anterior de la corriente de datos del mensaje.

No puede añadir la recuperación tras desastre a un gestor de colas, aunque puede migrar un gestor de colas existente para que se convierta en un gestor de colas RDQM (consulte “Migración de un gestor de colas para que se convierta en un gestor de colas DR RDQM” en la página 633).

Puede tener varios pares de gestores de colas de RDQM en ejecución en varios servidores distintos. Por ejemplo, podría tener gestores de colas de recuperación tras desastre primarios que se ejecutan en nodos diferentes, mientras que todos sus gestores de colas de recuperación tras desastre secundarios se ejecutan en el mismo nodo. En los diagramas siguientes se muestran algunas configuraciones de ejemplo.

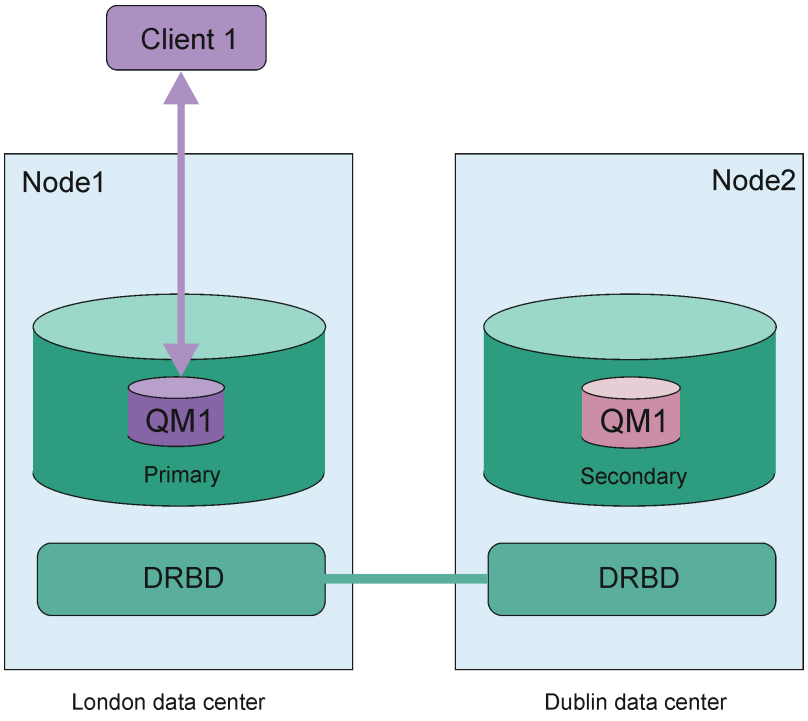


Figura 81. Par de RDQM único

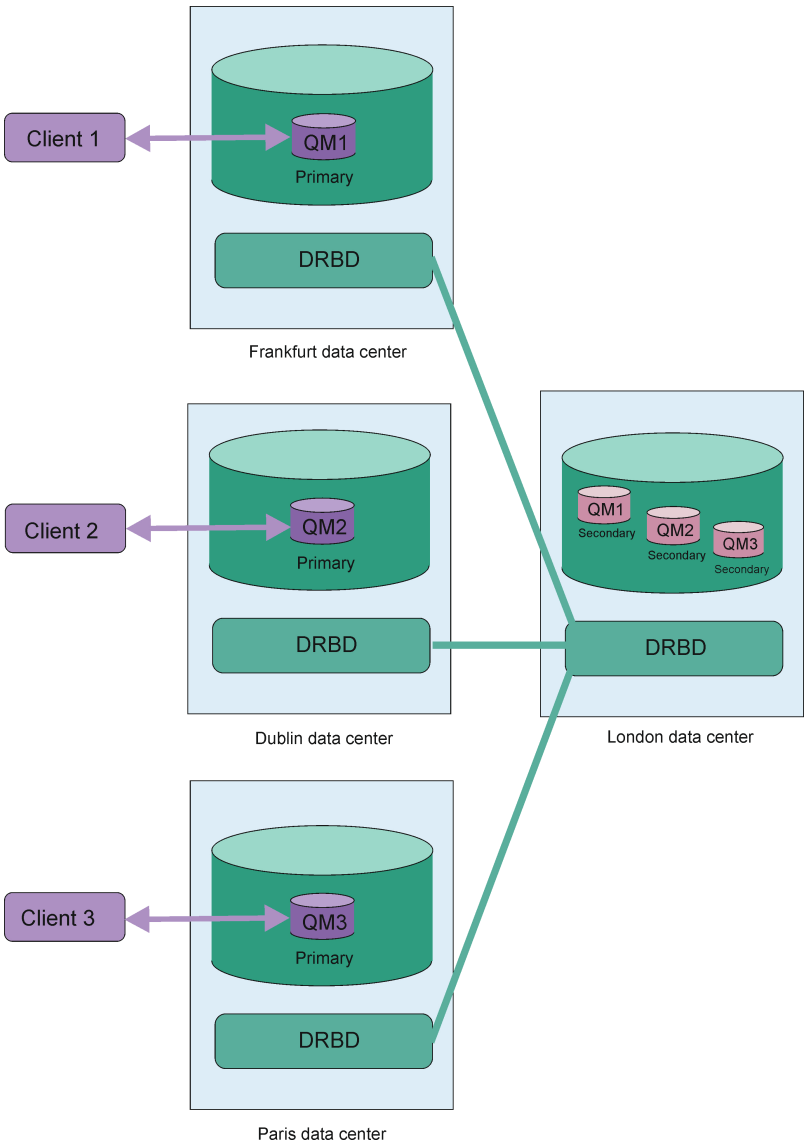


Figura 82. Gestores de colas secundarios en el mismo nodo

Réplica, sincronización e instantáneas

Aunque los dos nodos de una configuración de recuperación tras desastre están conectados, las actualizaciones en los datos persistentes de un gestor de colas de recuperación tras desastre se transmiten de la instancia primaria del gestor de colas a la instancia secundaria. Esto se conoce como **réplica**.

Si se pierde la conexión de red entre los dos nodos, se realizará un seguimiento de los cambios en los datos persistentes para la instancia primaria de un gestor de colas. Cuando la conexión de red se restaura, se utiliza un proceso distinto para poner al día la instancia secundaria lo más rápidamente posible. Esto se conoce como **sincronización**.

Mientras la sincronización está en curso, los datos en la instancia secundaria están en un estado incoherente. Se toma una **instantánea** del estado de los datos del gestor de colas secundario. Si durante la sincronización se produce un error del nodo principal o de la conexión de red, la instancia secundaria vuelve a esta instantánea y el gestor de colas se puede iniciar. No obstante, se pierden las actualizaciones que se han producido desde el error de red original.

Datos de partición (cerebro dividido)

Las configuraciones de RDQM de DR requieren la acción del usuario de la instancia primaria después de la pérdida de la instancia primaria de un gestor de colas para promocionar y ejecutar la instancia secundaria en el nodo de recuperación. Es responsabilidad de cualquiera que (o lo que sea) promueva la instancia secundaria para asegurarse de que el gestor de colas primario anterior se ha detenido. Si el primario original se sigue ejecutando, podría procesar mensajes y, cuando se restaura el funcionamiento normal, las dos instancias del gestor de colas tienen vistas de datos diferentes. Esto se conoce como estado particionado o de cerebro dividido.

Considere las situaciones siguientes:

- El nodo en el cual el gestor de colas primario se está ejecutando falla por completo. Debe promocionar la instancia secundaria para que se convierta en la instancia primaria; no puede realizar la acción para detener la instancia primaria, porque no se está ejecutando. Cuando el nodo original se repara o se sustituye, el gestor de colas en dicho nodo se convertirá inicialmente en el secundario y se sincronizará con el gestor de colas primario en el nodo de recuperación. Los roles de los dos gestores de colas se invierten, y el funcionamiento normal vuelve a comenzar. La única posible pérdida de datos en esta situación es cualquier dato que el primario no haya completado de duplicar en el secundario antes de que falle el nodo.
- Hay una anomalía de red que afecta al enlace de réplica entre los nodos que ejecutan las instancias primaria y secundaria del gestor de colas. En esta situación, debe asegurarse de que detiene la primaria original antes de promocionar la secundaria. Si la primaria original sigue teniendo otra conectividad de red, efectivamente tiene dos instancias primarias que se ejecutan a la vez, y se pueden acumular datos particionados. (Si el enlace de réplica está funcionando, no puede promocionar un gestor de colas secundario, si la instancia primaria sigue en ejecución, el mandato falla.)
- Hay una anomalía de red en el nodo que ejecuta la instancia primaria del gestor de colas. De nuevo, debe asegurarse de que detiene la instancia primaria antes de promocionar el secundario. Si la primaria anterior se sigue ejecutando cuando se restaura la red, habrá dos instancias primarias y, de nuevo, se acumularán datos particionados.

Cuando realizar una migración tras error gestionada, no verá un estado de DR de `partitioned` para las instancias del gestor de colas. Una migración tras error gestionada finaliza el gestor de colas en el nodo primario y, a continuación, inicia el gestor de colas en el nodo de recuperación, después de que los datos se hayan duplicado por completo. No se espera un estado particionado porque el gestor de colas se ha finalizado y se han sincronizado los datos entre los nodos antes que se inicie en el nodo de recuperación. Si el gestor de colas se inicia en el nodo de recuperación mientras hay una pérdida de conectividad entre los nodos, es probable que se produzca una divergencia de datos, si el gestor de colas estaba activo en el nodo principal cuando se perdió la conectividad. En este escenario, se espera que se notifique un estado particionado, una vez que se haya restaurado la conectividad porque los datos del gestor de colas no estaban sincronizados. Si se produce un estado particionado, es posible que tenga que examinar los dos conjuntos de datos y tomar una decisión informada sobre qué conjunto conservar. Consulte “Resolución de un problema particionado (cerebro divido) en DR RDQM” en la página 649.

Linux *Requisitos de la solución de RDQM de DR*

Antes de configurar un par de gestores de colas de recuperación tras desastre (DR) de RDQM, debe cumplir una serie de requisitos.

Requisitos del sistema

Antes de configurar la DR de RDQM, debe completar algunas tareas de configuración en cada uno de los servidores que alojarán los gestores de colas de DR del RDQM.

- Cada nodo requiere un grupo de volúmenes denominado `drbdpool`. El almacenamiento para cada gestor de colas de datos replicados de recuperación tras desastre (RDQM de DR) se asigna como dos volúmenes lógicos distintos por gestor de colas desde este grupo de volúmenes. (Cada gestor de colas requiere dos volúmenes lógicos para permitir la reversión a la operación de instantánea, de manera que se asigna a cada RDQM de DR apenas el doble del almacenamiento que especifica al crearlo.) Para

obtener el mejor rendimiento, este grupo de volúmenes tiene que constar de uno o varios volúmenes físicos que se correspondan con unidades de disco internas (preferiblemente SSD).

- Después de haber creado el grupo de volúmenes `drbdpool`, no haga nada más con él. IBM MQ gestiona los volúmenes lógicos creados en `drbdpool` y cómo y dónde se montan.
- Cada nodo requiere una interfaz que se utiliza para la réplica de datos. Esta debería tener un ancho de banda suficiente para soportar los requisitos de réplica dada la carga de trabajo que se espera de todos los gestores de colas de datos replicados.

  Para maximizar la tolerancia a errores, esta interfaz debería ser una tarjeta de interfaz de red (NIC) independiente.
- DRBD requiere que cada nodo utilizado para RDQM tenga un nombre de host de Internet válido (el valor devuelto por `uname -n`), tal como se define en RFC 952 modificado por RFC 1123.
- Si hay un cortafuegos entre los nodos utilizados para RDQM de DR, el cortafuegos debe permitir el tráfico entre los nodos en los puertos utilizados para la réplica. Se proporciona un script de ejemplo, `/opt/mqm/samp/rdqm/firewalld/configure.sh`, que abre los puertos necesarios si está ejecutando el cortafuegos estándar en RHEL. Debe ejecutar el script como `root`. Si está usando algún otro cortafuegos, examine las definiciones de servicio `/usr/lib/firewalld/services/rdqm*` para ver qué puertos hay que abrir. El script añade las reglas de servicio firewallD permanentes siguientes para DRBD e IBM MQ (puede editar el script para omitir los puertos Pacemaker si no utiliza HA):
  - MQ_INSTALLATION_PATH/samp/rdqm/firewalld/services/rdqm-drbd.xml permite los puertos TCP 7000 a 7100.
  - MQ_INSTALLATION_PATH/samp/rdqm/firewalld/services/rdqm-mq.xml permite el puerto TCP 1414 (debe editar el script si necesita un puerto distinto)
- Si el sistema utiliza SELinux en una modalidad que no sea la permisiva, debe ejecutar el mandato siguiente:

```
semanage permissive -a drbd_t
```

Requisitos de red

Se recomienda que localice los nodos utilizados para la recuperación tras desastre en distintos centros de datos.

Debe tener en cuenta las siguientes limitaciones:

- El rendimiento disminuye rápidamente con una mayor latencia entre centros de datos. IBM dará soporte a una latencia de hasta 5 ms para la réplica síncrona y 100 ms para la réplica asíncrona.
- Los datos enviados mediante el enlace de réplica no está sujetos a ningún otro cifrado aparte del que ya se pueda aplicar al utilizar IBM MQ AMS.
- La configuración de un gestor de colas RDQM para la recuperación tras desastre incurre en una sobrecarga debido a la necesidad de replicar datos entre dos nodos RDQM. La réplica síncrona incurre en una mayor sobrecarga que la réplica asíncrona. Cuando se utiliza la réplica síncrona, las operaciones de E/S del disco se bloquean hasta que se han escrito los datos en ambos nodos. Cuando se utiliza la réplica asíncrona, los datos se deben escribir solo en el nodo primario antes de que el proceso pueda continuar.

Requisitos de usuario para trabajar con gestores de colas

Para crear, suprimir o configurar gestores de colas de datos replicados (RDQM), debe ser el usuario root, o bien tener un ID de usuario perteneciente al grupo `mqm`, que tiene otorgado autorización sudo para los mandatos siguientes:

- **`crtmqm`**
- **`dltmqm`**
- **`rdqmdr`**

Un usuario perteneciente al grupo mqm puede ver el estado de un RDQM de DR mediante los mandatos siguientes:

- **dspmq**
- **rdqmstatus**

El usuario de mqm debe tener el mismo UID en ambos servidores y el grupo mqm debe tener el mismo GID en ambos servidores.

Linux *Creación de un RDQM de recuperación tras desastre*

El mandato **crtmqm** se utiliza para crear un gestor de colas de datos replicados (RDQM) para que funcione como primario o secundario en una configuración de recuperación tras desastre.

Acerca de esta tarea

Se puede crear un gestor de colas de datos replicados (RDQM) como un usuario del grupo mqm si dicho usuario puede utilizar sudo. De lo contrario, debe crear el RDQM como root.

Debe crear un gestor de colas de RDQM de DR primario en un solo nodo. A continuación, debe crear una instancia secundaria del mismo gestor de colas en otro nodo. Las instancias primaria y secundaria deben tener el mismo nombre y tener asignada la misma cantidad de almacenamiento.

Los puntos siguientes proporcionan algunas directrices sobre el dimensionamiento del sistema de archivos del gestor de colas:

1. Al crear un gestor de colas RDQM, se asigna un sistema de archivos para almacenar datos y registros del gestor de colas. Es importante dimensionar este sistema de archivos de forma adecuada para que el gestor de colas pueda registrar la actividad en curso en sus registros y almacenar los mensajes de aplicación en las colas. Al dimensionar el sistema de archivos, tenga en cuenta los requisitos máximos de mensajería, el crecimiento futuro de la carga de trabajo y las paradas de las aplicaciones que pueden hacer que los mensajes se acumulen en las colas. Para obtener instrucciones sobre cómo calcular el tamaño del registro de recuperación del gestor de colas, consulte “¿Qué tamaño debe tener el sistema de archivos de registro?” en la página 685. Al calcular los requisitos de almacenamiento para los mensajes de aplicación, es necesario tener en cuenta el tamaño y el número de mensajes, además de su cabecera MQMD y cualquier propiedad de mensaje que tengan.
2. Los sistemas de archivos del gestor de colas RDQM no se pueden redimensionar dinámicamente. Debe realizar una copia de seguridad y, a continuación, restaurar un gestor de colas RDQM con un sistema de archivos más grande si es necesario, consulte “Redimensionar el sistema de archivos para un gestor de colas RDQM HA” en la página 611.
3. Puede limitar el tamaño de colas individuales en disco utilizando atributos de cola local, como MAXDEPTH y MAXFSIZE. Consulte Modificación de archivos de cola de IBM MQ.
4. Debe supervisar el uso de disco en curso y responder adecuadamente si el uso de disco aumenta antes de que el uso del sistema de archivos pase a ser crítico. El uso del sistema de archivos se puede supervisar utilizando las prestaciones de plataforma/sistema operativo o suscribiéndose a las métricas publicadas en los temas del sistema IBM MQ que se describen en Métricas publicadas en los temas del sistema.

Procedimiento

- Para crear un RDQM de DR primario:
  a) Escriba el mandato siguiente:

```
crtmqm -rr p [-rt (a | s)] -rl Local_IP -ri Recovery_IP -rn Recovery_Name -rp Port
[other_crtmqm_options] [-fs size] QMname
```

  donde:

  -rr p
  Especifica que está creando la instancia primaria del gestor de colas.

-rt a | s
-rt s especifica que la configuración de DR utiliza la réplica síncrona, **-rt a** especifica que la configuración de DR utiliza la réplica asíncrona. La réplica asíncrona es el valor predeterminado.

-rl *IP_local*
Especifica que se utilizará la dirección IP para la réplica de DR de este gestor de colas.

-ri *IP_recuperación*
Especifica la dirección IP de la interfaz utilizada para la réplica en el servidor que aloja la instancia secundaria del gestor de colas.

-rn *nombre_recuperación*
Especifica el nombre del sistema que aloja la instancia secundaria del gestor de colas. El nombre es el valor que se devuelve si se ejecuta `uname -n` en ese servidor. Debe crear explícitamente un gestor de colas secundario en ese servidor.

-rp *Puerto*
Especifica el puerto que se utilizará para la réplica de DR.

otras_opciones_crtmqm
Puede especificar opcionalmente una o varias de estas opciones generales de **crtmqm**:

- -z
- -q
- -c *Texto*
- -d *ColaTransmisiónPredeterminada*
- -h *MaxManejadores*
- -g *GrupoAplicaciones*
- -oa *usuario*|*grupo*
- -t *TrigInt*
- -u *ColaMsjNoEntregados*
- -x *MaxMsjU*
- -lp *RegPri*
- -ls *RegSec*
- -lc | -l
- -lla | -lln
- -lf *TamañoArchivoRegistro*
- -p *Puerto*

-fs *tamaño*
De forma opcional, especifica el tamaño del sistema de archivos para crear el gestor de colas, es decir, el tamaño del volumen lógico que se crea en el grupo de volúmenes drbdpool. También se crea otro volumen lógico de ese tamaño, para permitir la reversión a la operación de instantánea, de manera que el almacenamiento total del RDQM de DR es apenas el doble que el especificado aquí.

Tamaño es un valor numérico, que se especifica en GB. Puede especificar un valor en MB especificando el valor seguido del carácter `M`. Por ejemplo, para especificar un tamaño de sistema de archivos de 3 GB, especifique 3. Para especificar un tamaño de sistema de archivos de 1024 MB, especifique 1024M. (También puede añadir un sufijo G para indicar explícitamente GB.)

nombreGC
Especifica el nombre del gestor de colas de datos replicados. El nombre es sensible a las mayúsculas y minúsculas.

Una vez que se completa el mandato, genera el mandato que se debe especificar en el nodo secundario para crear la instancia secundaria del gestor de colas. También puede utilizar el mandato **rdqmdr** en el nodo primario para recuperar el mandato **crtmqm** que necesita ejecutar

en el nodo secundario para crear el gestor de colas secundario, consulte "Gestión de las características primarias y secundarias de los RDQM de DR" en la página 639.

- Para crear un RDQM de DR secundario:
  a) Especifique el mandato siguiente en el nodo que alojará las instancias secundarias del RDQM:

```
crtmqm  -rr s [-rt (a | s)] -rl Local_IP -ri Primary_IP -rn Primary_Name  -rp Port
[other_crtmqm_options] [-fs size] QMname
```

  Donde:

  -rr s
  Especifica que está creando la instancia secundaria del gestor de colas.

  -rt a | s
  `-rt s` especifica que la configuración de DR utiliza la réplica síncrona, **`-rt a`** especifica que la configuración de DR utiliza la réplica asíncrona.

  -rl *IP_local*
  Especifica que se utilizará la dirección IP para la réplica de DR de este gestor de colas.

  -ri *IP_primaria*
  Especifica la dirección IP de la interfaz utilizada para la réplica en el servidor que aloja la instancia primaria del gestor de colas.

  -rn *Nombre_Primario*
  Especifica el nombre del sistema que aloja la instancia primaria del gestor de colas. El nombre es el valor que se devuelve si se ejecuta `uname -n` en ese servidor.

  -rp *Puerto*
  Especifica el puerto que se utilizará para la réplica de DR.

  otras_opciones_crtmqm
  Puede especificar opcionalmente una o varias de estas opciones generales de **`crtmqm`**:

  – `-z`

  -fs *tamaño*
  Especifica el tamaño del sistema de archivos que se debe crear para el gestor de colas, es decir, el tamaño del volumen lógico que se crea en el grupo de volúmenes drbdpool. Si ha especificado un tamaño no predeterminado al crear el gestor de colas primario, debe especificar el mismo valor aquí.

  Tamaño es un valor numérico, que se especifica en GB. Puede especificar un valor en MB especificando el valor seguido del carácter `M`. Por ejemplo, para especificar un tamaño de sistema de archivos de 3 GB, especifique 3. Para especificar un tamaño de sistema de archivos de 1024 MB, especifique 1024M. (También puede añadir un sufijo G para indicar explícitamente GB.)

  nombreGC
  Especifica el nombre del gestor de colas de datos replicados. Debe ser el nombre especificado para la instancia primaria del gestor de colas. Tenga en cuenta que el nombre es sensible a mayúsculas y minúsculas.

Qué hacer a continuación

Cuando haya creado las instancias primaria y secundaria del gestor de colas, debe comprobar el estado en ambos nodos, para verificar si son correctos. Utilice el mandato **`rdqmstatus`** en ambos nodos. Los nodos deben mostrar el estado normal, tal como se describe en "Visualización del estado de RDQM de DR" en la página 641. Si no muestran ese estado, suprima la instancia secundaria y vuelva a crearla, prestando atención a utilizar los argumentos correctos.

Referencia relacionada

crtmqm

Linux *Supresión de un RDQM de DR*

El mandato **dltmqm** se utiliza para suprimir un gestor de colas de datos replicados de recuperación tras desastre (RDQM).

Acerca de esta tarea

Debe ejecutar el mandato para suprimir el RDQM en ambos nodos del RDQM, primario y secundario. Antes hay que parar el RDQM. El mandato se puede ejecutar como usuario mqm si dicho usuario tiene los privilegios sudo necesarios. De lo contrario, hay que ejecutar el mandato como root.

Procedimiento

- Para suprimir un RDQM de DR, especifique el mandato siguiente:

```
dltmqm RDQM_name
```

Referencia relacionada

dltmqm

MQ Adv. Linux *Migración de un gestor de colas para que se convierta en un gestor de colas DR RDQM*

Puede migrar un gestor de colas existente para que se convierta en el gestor de colas de datos replicados (RDQM) de recuperación tras desastre (DR) haciendo una copia de seguridad de sus datos persistentes y, después, restaurando los datos en un gestor de colas RDQM recién creado que tenga el mismo nombre.

Acerca de esta tarea

Los gestores de colas de datos replicados de DR requieren un volumen lógico dedicado (sistema de archivos) y la configuración de la réplica de disco. Estos componentes solo se configuran cuando se crea un nuevo gestor de colas. Un gestor de colas existente se puede migrar para utilizar el RDQM haciendo una copia de seguridad de sus datos persistentes y, después, restaurando los datos en un gestor de colas RDQM recién creado que tenga el mismo nombre. Este procedimiento conserva la configuración del gestor de colas, el estado y los mensajes persistentes en el momento en que se creó la copia de seguridad.

Nota: Solo puede migrar un gestor de colas de una versión de IBM MQ que sea igual o inferior a la versión en la que se ha instalado el RDQM. El sistema operativo y la arquitectura también debe ser los mismos. De lo contrario, debe crear un nuevo gestor de colas en la plataforma de destino, consulte Trasladar un gestor de colas a un sistema operativo diferente.

Antes de migrar un gestor de colas, debe cumplir las condiciones siguientes:

- Evalúe los requisitos del a recuperación tras desastre y consulte “Recuperación tras desastre de RDQM” en la página 626.
- Revise las aplicaciones y los gestores de colas que se conectan al gestor de colas. Considere los cambios necesarios para direccionar las conexiones al nodo RDQM donde se está ejecutando el gestor de colas.
- Suministre, o identifique, los nodos RDQM existentes para la configuración que seleccione. Para obtener más información sobre los requisitos del sistema para el RDQM, consulte “Requisitos de la solución de RDQM de DR” en la página 628.
- Instale IBM MQ Advanced, que incluye la característica RDQM, en cada nodo.
- Si lo desea, verifique la configuración de RDQM utilizando un gestor de colas de prueba, que después se puede suprimir. Se recomienda probar la configuración para identificar y resolver los problemas antes de migrar el gestor de colas.
- Revise la configuración de seguridad para el gestor de colas y, a continuación, duplique los grupos y usuarios locales necesarios en cada nodo RDQM.

- Revise el gestor de colas y la configuración de canal para determinar si se utilizan salidas de API, salidas de canal o salidas de conversión de datos. Instale las salidas necesarias en cada nodo RDQM.
- Revise los servicios del gestor de colas que se han definido y, a continuación, instale y configure los procesos necesarios en cada nodo RDQM.

Procedimiento

1. Haga una copia de seguridad del gestor de colas existente:
   a) Detenga el gestor de colas existentes emitiendo un mandato de conclusión en espera `endmqm -w`, o un mandato de conclusión inmediata `endmqm -i`. Este paso es importante para garantizar que los datos de la copia de seguridad son coherentes.
   b) Determine la ubicación del directorio de datos del gestor de colas visualizando el archivo de configuración de IBM MQ, `mqs.ini`. En Linux, este archivo se encuentra en el directorio `/var/mqm`. Para obtener más información sobre `mqs.ini`, consulte "Archivo de configuración de IBM MQ, mqs.ini" en la página 96.

      Localice la stanza `QueueManager` para el gestor de colas en el archivo. Si la stanza contiene una clave llamada `DataPath`, su valor es el directorio de datos del gestor de colas. Si la clave no existe, el directorio de datos del gestor de colas se puede determinar utilizando los valores de las claves `Prefix` y `Directory`. El directorio de datos del gestor de colas es una concatenación de estos valores, con el formato *prefijo*/qmgrs/*directorio*. Para obtener más información sobre la stanza QueueManager, consulte "Stanza QueueManager del archivo mqs.ini" en la página 107.
   c) Cree una copia de seguridad del directorio de datos de gestor de colas. En Linux, puede hacerlo utilizando el mandato **tar**. Por ejemplo, para hacer una copia de seguridad del directorio de datos para un gestor de colas, puede utilizar el mandato siguiente. Observe el último parámetro del mandato, que es un único punto:

      ```
      tar -cvzf qm-data.tar.gz -C queue_manager_data_dir .
      ```

   d) Determine la ubicación del directorio de registro del gestor de colas visualizando el archivo de configuración del gestor de colas de IBM MQ `qm.ini`. Este archivo se encuentra en el directorio de datos del gestor de colas. Para obtener más información sobre el archivo, consulte "Archivos de configuración de gestores de colas, qm.ini" en la página 109.

      El directorio de registro del gestor de colas se define como el valor de la clave `LogPath` en la stanza `Log`. Para obtener información sobre la stanza, consulte "Stanza de registro del archivo qm.ini" en la página 145.
   e) Cree una copia de seguridad del directorio de registro del gestor de colas. En Linux, puede hacer esto utilizando el mandato tar. Por ejemplo, para hacer una copia de seguridad del directorio de registro para un gestor de colas, puede utilizar el mandato siguiente. Observe el último parámetro del mandato, que es un único punto:

      ```
      tar -cvzf qm-log.tar.gz -C queue_manager_log_dir .
      ```

   f) Cree una copia de seguridad de los repositorios de certificados utilizados por el gestor de colas, si no se encuentran en el directorio de datos del gestor de colas. Asegúrese de que se ha hecho una copia de seguridad de ambos archivos, el archivo de base de datos de claves y el archivo de ocultación de contraseña. Para obtener información sobre el repositorio de claves del gestor de colas, consulte El repositorio de claves SSL/TLS y Ubicación del repositorio de claves para un gestor de colas. Para obtener más información sobre cómo localizar el almacén de claves AMS, si el gestor de colas se ha configurado para utilizar la intercepción del agente de canal de mensajes (MCA) AMS, consulte Intercepción del agente de canal de mensajes (MCA).
   g) El gestor de colas existente ya no es necesario, así que se puede suprimir. Sin embargo, siempre que sea posible, solo debe suprimir el gestor de colas existente, después de que se haya restaurado correctamente en el sistema de destino. El aplazamiento de la supresión garantiza que el gestor de colas se puede reiniciar si el proceso de migración no se completa correctamente.

Nota: Si aplaza la supresión del gestor de colas existente, no lo reinicie. Es importante que el gestor de colas permanezca finalizado porque los cambios adicionales en si configuración o estado se pierden durante la migración.

2. Prepare el nodo RDQM primario:
   a) Cree un nuevo gestor de colas RDQM con el mismo nombre que el gestor de colas del que ha hecho una copia de seguridad. Asegúrese de que el sistema de archivos asignado para el gestor de colas RDQM por **crtmqm** es lo suficientemente grande para que contenga los datos, los registros primarios y los registros secundarios para el gestor de colas existente, además de algún espacio adicional para una futura ampliación. Para obtener información sobre cómo crear un gestor de colas RDQM, consulte "Creación de un RDQM de recuperación tras desastre" en la página 630.
   b) Determine el nodo RDQM primario para el gestor de colas. Para obtener más información sobre cómo determinar el nodo primario, consulte rdqmstatus (mostrar estado RDQM).
   c) En el nodo RDQM primario, si el gestor de colas RDQM se inicia, deténgalo utilizando el mandato `endmqm -w` o `endmqm -i`.
   d) Determine la ubicación de los directorios de datos y registro para el gestor de colas RDQM (utilice los métodos descritos en los pasos 1b y 1d).
   e) Suprima el contenido de los directorios de datos y registro del gestor de colas RDQM, pero no los propios directorios.
3. Restaure el gestor de colas en el nodo RDQM primario:
   a) Copie las copias de seguridad de los directorios de datos y registro del gestor de colas en el nodo RDQM primario, además de las copias de seguridad independientes de los repositorios de certificados utilizados por el gestor de colas.
   b) Restaure la copia de seguridad del directorio de datos del gestor de colas en el directorio de datos vacío para el nuevo gestor de colas RDQM, asegurándose de que se conservan los permisos y la propiedad de archivos. Si la copia de seguridad se ha creado utilizando el mandato tar de ejemplo en el paso 1c, el usuario root puede utilizar el mandato siguiente para restaurarla:

      ```
      tar -xvzpf qm-data.tar.gz -C queue_manager_data_dir
      ```

   c) Restaure la copia de seguridad del directorio de registro del gestor de colas en el directorio de registro vacío para el nuevo gestor de colas RDQM, asegurándose de que se conserven los permisos y la propiedad de archivos. Si la copia de seguridad se ha creado utilizando el mandato tar de ejemplo en el paso 1e, el usuario root puede utilizar el mandato siguiente para restaurarla:

      ```
      tar -xvzpf qm-log.tar.gz -C queue_manager_log_dir
      ```

   d) Edite el archivo de configuración del gestor de colas restaurado, `qm.ini`, en el directorio de datos del gestor de colas RDQM. Actualice el valor de la clave `LogPath` en la stanza `Log` para especificar el directorio de registro para el gestor de colas RDQM.

      Revise otras vías de acceso de archivo que están definidas en el archivo de configuración y actualícelas, si es necesario. Por ejemplo, es posible que tenga que actualizar las vías de acceso siguientes:

      - La vía de acceso de los archivos de registro de errores generados por los servicios de mensajes de diagnóstico.
      - La vía de acceso para las salidas necesarias para el gestor de colas.
      - La vía de acceso para los archivos de carga conmutada, si el gestor de colas es un coordinador de transacciones XA.

   e) Si el gestor de colas se ha configurado para utilizar la intercepción del agente de canal de mensajes (MCA) AMS, copie el almacén de claves AMS en la nueva instalación de RDQM y, después, revise y actualice la configuración. El almacén de claves debe estar disponible en cada nodo RDQM, de modo que si no se encuentra en el sistema de archivos duplicado para el gestor de colas, en su lugar, se debe copiar en cada nodo. Para obtener más información, consulte Intercepción del agente de canal de mensajes (MCA).

f) Verifique que el gestor de colas se muestra mediante el mandato **dspmq** y que su estado se notifica como finalizado. El ejemplo siguiente muestra una salida de ejemplo para un gestor de colas RDQM DR:

```
$ dspmq -o status -o dr

QMNAME(QM1) STATUS(Ended normally) DRROLE(Primary)
```

g) Verifique que los datos del gestor de colas restaurado se han duplicado en los nodos RDQM secundarios utilizando el mandato **rdqmstatus** para mostrar el estado del gestor de colas. El estado de DR se debe notificar como `Normal` en cada nodo. El ejemplo siguiente muestra una salida de ejemplo para un gestor de colas RDQM DR:

```
$ rdqmstatus -m QM1
Queue manager status:                   Ended normally
Queue manager file system:              51MB used, 1.0GB allocated [5%]
DR role:                                Primary
DR status:                              Normal
DR type:                                Synchronous
DR port:                                3000
DR local IP address:                    192.168.20.1
DR remote IP address:                   192.168.20.2
```

h) Inicie el gestor de colas en el nodo RDQM primario.

i) Conéctese al gestor de colas y actualice el valor del atributo del gestor de colas SSLKEYR para especificar la nueva ubicación del repositorio de certificados del gestor de colas. De forma predeterminada, el valor de este atributo se establece en *queue_manager_data_directory*`/ssl/key`. El repositorio de certificados debe estar ubicado en la misma ubicación en cada nodo RDQM. Si el repositorio no se encuentra en el sistema de archivos duplicado para el gestor de colas, en su lugar, se debe copiar en cada nodo.

j) Revise las definiciones de objeto de IBM MQ para el gestor de colas y actualice el valor de los atributos del objeto que hacen referencia a los valores de red cambiados, el directorio de instalación de IBM MQ o el directorio de datos del gestor de colas, incluidos los objetos siguientes:

- Direcciones IP locales utilizadas por escuchas (atributo `IPADDR`)
- Direcciones IP locales utilizadas por canales (atributo `LOCLADDR`)
- Direcciones IP locales definidas para los canales de clúster receptor (atributo `CONNAME`)
- Direcciones IP locales definidas para los objetos de información de comunicación (atributo `GRPADDR`)
- Vías de acceso de sistema definidas para las definiciones de objeto de proceso y servicio.

k) Detenga y reinicie el gestor de colas para asegurarse de que los cambios acaben siendo efectivos.

l) Repita el paso 3j para los gestor de colas remotos. además de los valores equivalentes para las aplicaciones, que se conectan al gestor de colas migrados, incluyendo:

- Nombre de conexión de canal (atributo `CONNAME`)
- Reglas de autenticación de canal que restringen las conexiones de entrada del gestor de colas basándose en su dirección IP o nombre de host.
- Tablas de definición de canal de cliente (CCDT), valores de nombre de dominio (DNS), direccionamiento de red o información de conexión equivalente.

m) Realice una migración tras error gestionada del gestor de colas a cada nodo RDQM para garantizar que se ha establecido correctamente la configuración necesaria, consulte "Conmutación a un nodo de recuperación" en la página 646.

Redimensionar el sistema de archivos para un gestor de colas RDQM de DR

Para redimensionar el sistema de archivos para un gestor de colas de datos replicados (RDQM) de recuperación tras desastre (DR), haga una copia de seguridad de sus datos persistententes y, después, restaure los datos en un gestor de colas RDQM recién creado que tenga el mismo nombre, pero un sistema de archivos de un tamaño diferente.

Acerca de esta tarea

Los gestores de colas de datos duplicados de DR requieren un volumen lógico dedicado (sistema de archivos) y la configuración de la réplica de disco. Estos componentes solo se configuran cuando se crea un nuevo gestor de colas. El sistema de archivos no se puede redimensionar después de que haya sido creado porque debe tener el mismo tamaño en cada nodo. Para redimensionar el sistema de archivos para un gestor de colas de datos duplicados (RDQM) existente, puede hacer una copia de seguridad de sus datos persistentes y, a continuación, restaurar los datos en un gestos de datos RDQM recién creado que tenga el mismo nombre, pero un sistema de archivos de un tamaño diferente. Este procedimiento conserva la configuración del gestor de colas, el estado y los mensajes persistentes en el momento en que se creó la copia de seguridad.

Procedimiento

1. Haga una copia de seguridad del gestor de colas RDQM existente en el nodo RDQM primario:
   a) Determine el nodo RDQM primario para el gestor de colas. Para obtener más información sobre cómo determinar el nodo primario, consulte rdqmstatus (mostrar estado RDQM).
   b) En el nodo RDQM primario, si el gestor de colas RDQM se inicia, deténgalo utilizando el mandato **endmqm -w** o **endmqm -i**.
   c) Determine la ubicación del directorio de datos del gestor de colas visualizando el archivo de configuración de IBM MQ, `mqs.ini`. En Linux, este archivo se encuentra en el directorio `/var/mqm`. Para obtener más información sobre `mqs.ini`, consulte “Archivo de configuración de IBM MQ, mqs.ini” en la página 96.

      Localice la stanza `QueueManager` para el gestor de colas en el archivo. El directorio de datos del gestor de datos es el valor de la clave llamada `DataPath`. Para obtener más información sobre la stanza `QueueManager`, consulte “Stanza QueueManager del archivo mqs.ini” en la página 107.
   d) Cree una copia de seguridad del directorio de datos de gestor de colas. En Linux, puede hacerlo utilizando el mandato **tar**. Por ejemplo, para hacer una copia de seguridad del directorio de datos para un gestor de colas, puede utilizar el mandato siguiente. Observe el último parámetro del mandato, que es un único carácter de punto (.):

      ```
      tar -cvzf qm-data.tar.gz -C queue_manager_data_dir .
      ```
   e) Determine la ubicación del directorio de registro del gestor de colas visualizando el archivo de configuración del gestor de colas de IBM MQ `qm.ini`. Este archivo se encuentra en el directorio de datos del gestor de colas. Para obtener más información sobre el archivo, consulte “Archivos de configuración de gestores de colas, qm.ini” en la página 109.

      El directorio de registro del gestor de colas se define como el valor de la clave `LogPath` en la stanza Log. Para obtener información sobre la stanza, consulte “Stanza de registro del archivo qm.ini” en la página 145.
   f) Cree una copia de seguridad del directorio de registro del gestor de colas. En Linux, puede hacerlo utilizando el mandato **tar**. Por ejemplo, para hacer una copia de seguridad del directorio de registro para un gestor de colas, puede utilizar el mandato siguiente. Observe el último parámetro del mandato, que es un único carácter de punto (.):

      ```
      tar -cvzf qm-log.tar.gz -C queue_manager_log_dir .
      ```
   g) Suprima el gestor de colas RDQM existente.
2. Restaure el gestor de colas con un sistema de archivos del tamaño necesario:
   a) Cree un nuevo gestor de colas RDQM con el mismo nombre que el gestor de colas del que ha hecho una copia de seguridad. Asegúrese de que el sistema de archivos asignado para el gestor de colas RDQM por **crtmqm** es el tamaño que requiere, y es lo suficientemente grande para contener los datos, registros primarios y registros secundarios para el gestor de colas existente, además de algo de espacio adicional para la futura expansión. Para obtener información sobre cómo crear un gestor de colas RDQM, consulte “Creación de un RDQM de recuperación tras desastre” en la página 630.

b) Determine el nodo RDQM primario para el gestor de colas. Para obtener más información sobre cómo determinar el nodo primario, consulte rdqmstatus (mostrar estado RDQM).

c) En el nodo RDQM primario, si se ha iniciado el gestor de colas RDQM, deténgalo utilizando el mandato **endmqm -w** o **endmqm -i** .

d) En el nodo RDQM primario, determine la nueva ubicación de los datos y los directorios de registro para el gestor de colas RDQM (utilice los métodos descritos en los pasos 1c y 1e).

e) En el nodo RDQM primario, suprima el contenido de los directorios de datos y registros del gestor de colas RDQM, pero no los propios directorios.

f) En el nodo RDQM primario, restaure la copia de seguridad del directorio de datos del gestor de colas en el directorio de datos vacío para el nuevo gestor de colas RDQM, asegurándose de que se conserven los permisos y la propiedad de archivos. Si la copia de seguridad se ha creado utilizando el mandato **tar** de ejemplo en el paso 1d, el usuario root puede utilizar el mandato siguiente para restaurarla:

```
tar -xvzpf qm-data.tar.gz -C queue_manager_data_dir
```

g) En el nodo RDQM primario, restaura la copia de seguridad del directorio de registro del gestor de colas en el directorio de registro vacío para el nuevo gestor de colas RDQM, asegurándose de que se conserven los permisos y la propiedad de archivos. Si la copia de seguridad se ha creado utilizando el mandato **tar** de ejemplo en el paso 1f, el usuario root puede utilizar el mandato siguiente para restaurarla:

```
tar -xvzpf qm-log.tar.gz -C queue_manager_log_dir
```

h) En el nodo RDQM primario, edite el archivo de configuración del gestor de colas restaurado, `qm.ini`, en el directorio de datos del nuevo gestor de colas RDQM. Actualice el valor de la clave `LogPath` en la stanza `Log` para especificar el directorio de registro para el nuevo gestor de colas RDQM que ha determinado en el paso 2d. Revise otras vías de acceso de archivo que están definidas en el archivo de configuración y actualícelas, si es necesario. Por ejemplo, es posible que tenga que actualizar las vías de acceso siguientes:

- La vía de acceso de los archivos de registro de errores generados por los servicios de mensajes de diagnóstico.
- La vía de acceso para las salidas necesarias para el gestor de colas.
- La vía de acceso para los archivos de carga conmutada, si el gestor de colas es un coordinador de transacciones XA.

i) Verifique que el gestor de colas se visualiza mediante el mandato **dspmq** y que su estado se notifica como ended. El ejemplo siguiente muestra una salida de ejemplo para un gestor de colas RDQM DR:

```
$ dspmq -o status -o dr
QMNAME(QM1) STATUS(Ended normally) DR(Primary)
```

j) Verifique que los datos del gestor de colas restaurados se han duplicado en el nodo RDQM secundario utilizando el mandato **rdqmstatus** para mostrar el estado para el gestor de colas. El estado de DR se debe notificar como `Normal` en cada nodo. El ejemplo siguiente muestra una salida de ejemplo para un gestor de colas RDQM DR en el nodo primario:

```
$ rdqmstatus -m QM1
Queue manager status:                   Running
CPU:                                    0.00
Memory:                                 123MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
DR role:                                Primary
DR status:                              Normal
DR type:                                Synchronous
DR port:                                3000
DR local IP address:                    192.168.20.1
DR remote IP address:                   192.168.20.2
```

El ejemplo siguiente muestra una salida de ejemplo para un gestor de colas RDQM DR en el nodo de recuperación:

```
Queue manager status:                   Ended immediately
DR role:                                Secondary
DR status:                              Normal
DR port:                                3000
DR local IP address:                    192.168.20.2
DR remote IP address:                   192.168.20.1
```

k) Inicie el gestor de colas en el nodo RDQM primario.

l) Realice una conmutación del gestor de colas al nodo de recuperación para asegurarse de que se haya establecido la configuración necesaria correctamente, consulte "Conmutación a un nodo de recuperación" en la página 646.

Almacenamiento del estado de aplicación persistente
Puede almacenar información de estado persistente relacionada con aplicaciones junto con los datos de gestor de colas.

Cada gestor de colas de IBM MQ tiene un sistema de archivos dedicado para su estado persistente, que incluye tanto sus datos de cola como el registro de recuperación. En una configuración RDQM un volumen lógico que se replica entre los sistemas Linux (nodos) respalda el sistema de archivos. El sistema de archivos incluye un directorio `userdata` que puede utilizar para almacenar información de estado persistente para las aplicaciones. Por lo tanto, cuando un gestor de colas de datos duplicados se traslada para ejecutarse en otro nodo de la configuración RDQM, tiene disponible el contexto de aplicación, así como el contexto del gestor de colas. Consulte Contenido de directorio en sistemas Unix y Linux.

Si elige almacenar el estado de aplicación en el directorio `userdata`, debe tener en cuenta que los datos escritos en esta ubicación pueden consumir el espacio de disco disponible asignado al gestor de colas. Debe asegurarse de que haya suficiente espacio de disco disponible para que el gestor de colas escriba datos de cola, registros y otra información de estado persistente.

El directorio `userdata` tiene la propiedad de usuario y grupo mqm y lo puede leer todo el mundo para que los usuarios puedan acceder al mismo sin necesidad de pertenecer al grupo de administradores de IBM MQ (es decir, mqm). No puede modificar los permisos del directorio `userdata`, pero puede crear contenido en él con la propiedad y los permisos necesarios.

Durante la migración tras error del gestor de colas RDQM, finaliza el gestor de colas y se desmonta su sistema de archivos en el nodo RDQM actual. Después, se monta el sistema de archivos y se reinicia el gestor de colas en otro nodo de la configuración RDQM. No se puede desmontar un sistema de archivos si un proceso tiene un manejador abierto para uno de sus archivos. Para asegurarse de que se puede completar una migración tras error del gestor de colas, si no se puede desmontar el sistema de archivos del gestor de colas, se envía una señal SIGTERM a los procesos que tienen un manejador de archivos abierto, seguida de una señal SIGKILL si no se han liberado los manejadores abiertos. Las aplicaciones deben estar diseñadas para responder correctamente a SIGTERM. Si las aplicaciones o los procesos se configuran como un servicio de gestor de colas, durante una migración tras error gestionada, se pueden finalizar durante la conclusión del gestor de colas antes de que se desmonte el sistema de archivos. Si una aplicación o un proceso no está configurado como un servicio de gestor de colas o se produce una migración tras error no gestionada, como por ejemplo una pérdida de quórum, es probable que se envíen señales para liberar el sistema de archivos.

Linux *Gestión de las características primarias y secundarias de los RDQM de DR*

Puede convertir un gestor de cola de datos replicados de recuperación tras desastre secundario (RDQM de DR) en un RDQM de DR primario. También puede convertir una instancia primaria en una instancia secundaria.

Acerca de esta tarea

Utilice el mandato **rdqmdr** para convertir una instancia secundaria de un RDQM en la instancia primaria. Es posible que deba completar esta acción si por algún motivo pierde la instancia primaria. A continuación, puede iniciar el gestor de colas y continuar ejecutándolo en el nodo de recuperación.

También puede utilizar el mandato **rdqmdr** para convertir una instancia primaria de un RDQM en la instancia secundaria. Es posible que deba completar esta acción, por ejemplo, si estaba reconfigurando el sistema.

También puede utilizar el mandato **rdqmdr** en un gestor de colas primario para recuperar el mandato exacto que necesita para crear una instancia secundaria de ese gestor de colas en el nodo de recuperación.

Puede utilizar el mandato **rdqmdr** como usuario del grupo `mqm` si el usuario puede utilizar sudo. De lo contrario, debe iniciar sesión como root.

Procedimiento

- Para convertir una instancia secundaria de un RDQM de DR en una instancia primaria, especifique el mandato siguiente:

  ```
  rdqmdr -m QMname -p
  ```

  Este mandato falla si la instancia primaria del gestor de colas aún se está ejecutando y el enlace de réplica de DR aún funciona.
- Para convertir una instancia primaria del gestor de colas en una instancia secundaria, especifique el mandato siguiente:

  ```
  rdqmdr -m QMname -s
  ```
- Para visualizar el mandato **crtmqm** necesario para configurar la instancia secundaria de un gestor de colas, especifique el mandato siguiente en el nodo primario:

  ```
  rdqmdr -d -m QMname
  ```

  Puede especificar el mandato **crtmqm** devuelto en el nodo secundario para crear la instancia secundaria del RD RDQM.

Linux *Inicio, detención y visualización del estado de un RDQM de DR*

Se usan variantes de mandatos de control de IBM MQ para iniciar, detener y ver el estado actual de un gestor de colas de datos replicados de recuperación tras desastre (RDQM de DR).

Acerca de esta tarea

Debe ejecutar los mandatos que inician, detienen y visualizan el estado actual de un gestor de colas de datos replicados (RDQM) como usuario perteneciente al grupo `mqm`.

Hay que ejecutar los mandatos para iniciar y parar un gestor de colas en el nodo primario de ese gestor de colas (es decir, el nodo en el que el gestor de colas está ejecutando actualmente).

Procedimiento

- Para iniciar un RDQM de DR, especifique el mandato siguiente en el nodo primario del RDQM:

  ```
  strmqm qmname
  ```

  donde *nombreGC* es el nombre del RDQM que se quiere iniciar.
- Para parar un RDQM, ejecute el mandato siguiente en el nodo primario del RDQM:

  ```
  endmqm qmname
  ```

  donde *nombreGC* es el nombre del RDQM que se desea parar.
- Para ver el estado un RDQM, ejecute el mandato siguiente:

  ```
  dspmq -m QMname
  ```

La información de estado que aparece en la salida depende de si se ejecuta el mandato en el nodo primario o secundario del RDQM. Si se ejecuta en el nodo primario, se mostrará uno de los mensajes de estado normal devueltos por **dspmq**. Si ejecuta el mandato en un nodo secundario, se muestra el estado `Ended immediately` . Por ejemplo, si se ejecuta **dspmq** en el nodo RDQM7, podría devolverse la siguiente información:

```
QMNAME(DRQM8)                             STATUS(Ended immediately)
QMNAME(DRQM7)                             STATUS(Running)
```

Puede utilizar argumentos con `dspmq` para establecer si un RDQM está configurado para la recuperación tras desastre, y si es actualmente la instancia primaria o secundaria:

```
dspmq -m QMname -o (dr | DR)
```

Se visualiza una de las respuestas siguientes:

DRROLE()
Indica que el gestor de colas no está configurado para la recuperación tras desastre.

DRROLE(Primary)
Indica que el gestor de colas está configurado como primario de DR.

DRROLE(Secondary)
Indica que el gestor de colas está configurado como secundario de DR.

Referencia relacionada

dspmq
endmqm
strmqm

Linux *Visualización del estado de RDQM de DR*

Puede ver el estado de todos los gestores de colas de datos replicados de recuperación tras desastre (RDQM de DR) en un nodo, o información detallada de un RDQM de DR especificado.

Acerca de esta tarea

El mandato **rdqmstatus** se utiliza para ver el estado de todos los RDQM de DR o de RDQM individuales.

El estado de resumen de un nodo también muestra información sobre el módulo de kernel de DRBD en el que se basa RDQM. Cuando se actualiza RDQM, es importante asegurarse de que se haya instalado la versión correcta del módulo de kernel de DRBD para la versión del kernel RHEL que se ejecuta en el sistema. El estado muestra la versión del kernel de sistema operativo, la versión del kernel para la que se ha creado el módulo DRBD, la versión de DRBD y el estado cargado del módulo de kernel de DRBD.

Debe ser un usuario del grupo `mqm` para ejecutar el mandato **rdqmstatus**. Puede ejecutar el mandato en cualquiera de los nodos del par de RDQM de DR.

Procedimiento

- Para ver el estado de resumen de todos los RDQM de DR de un nodo, ejecute el mandato siguiente en ese nodo:

  ```
  rdqmstatus
  ```

  Se visualiza el estado de los RDQM de DR en el nodo, por ejemplo:

  ```
  Node:                                 mqhavm07.exampleco.com
  OS kernel version:                    5.14.0-362.18.1
  DRBD OS kernel version:               5.14.0-362.18.1
  DRBD version:                         9.2.7
  DRBD kernel module status:            Loaded

  Queue manager name:                   DRQM8
  Queue manager status:                 Ended immediately
  ```

```
DR role:                                Secondary

Queue manager name:                     DRQM7
Queue manager status:                   Running
DR role:                                Primary
```

El estado del módulo de kernel de DRBD tiene uno de los siguientes valores:

Cargado
Indica que se ha cargado el módulo DRBD.

Cargado parcialmente
Puede producirse cuando se ha cargado el módulo DRBD, pero no funciona correctamente debido a una discrepancia.

No cargado

No se ha cargado el módulo DRBD. Esto se puede visualizar en una configuración recién instalada, cuando todavía no se ha creado ningún gestor de colas RDQM.

No instalado
Indica que el módulo DRBD no está instalado. o que IBM MQ no ha podido determinar la versión de kernel del sistema operativo del módulo DRBD.

Versión instalada anteriormente todavía cargada
Este estado puede producirse si se instala un nuevo módulo DRBD mientras se ejecuta el módulo DRBD existente (es decir, se está ejecutando un gestor de colas RDQM). El módulo recién instalado se notifica en el estado, pero no es el módulo que se está ejecutando realmente.

- Para ver el estado de un RDQM específico, especifique el mandato siguiente:

```
rdqmstatus -m  qmname
```

En la tabla siguiente se resume la información que se devuelve.

Tabla 35. Atributos de estado

Atributo de estado	**Valores posibles**	**Cuándo se muestra**
Estado del gestor de colas	estado (tal como lo visualiza `dspmq`)	Siempre se muestra
CPU	*n.nn*%	Solo se muestra cuando RDQM en el nodo actual tiene el rol primario
Memoria	*nnn*MB	Solo se muestra cuando RDQM en el nodo actual tiene el rol primario
Sistema de archivos del gestor de colas	*nnn*MB utilizados,*n.n*GB asignados [*n*%]	Solo se muestra cuando RDQM en el nodo actual tiene el rol primario
Rol de DR	Primario Secundario Desconocido	Siempre se muestra
Estado de DR	Normal	Operación normal
	Sincronización en curso	La sincronización está en curso
	Particionado	El gestor de colas se ha iniciado en ambos nodos mientras la red de réplica de DR no está disponible

Tabla 35. Atributos de estado (continuación)

Atributo de estado	Valores posibles	Cuándo se muestra
	Sistema remoto no disponible	Se ha perdido la conexión al otro nodo
	Incoherente	Había una sincronización en curso, pero se ha interrumpido
	Restaurando a la instantánea	El usuario ha optado por volver a la instantánea que se realizó cuando el gestor de colas entró en el estado Incoherente.
	Sistema remoto no configurado	Se ha configurado la instancia primaria del RDQM, pero no se ha configurado ninguna instancia secundaria
	Negociación fallida	Uno de los nodos se ha establecido en la réplica síncrona y el otro en la réplica asíncrona
Tipo de DR	Síncrono o asíncrono	Siempre se muestra
Puerto de DR	*número_puerto* (el puerto TCP/IP utilizado para replicar los datos para este gestor de colas)	Siempre se muestra
Dirección IP local de DR	La dirección IP local desde la que este gestor de colas realiza la réplica para DR	Siempre se muestra
Dirección IP remota de DR	La dirección IP remota a la que este gestor de colas realiza la réplica para DR	Siempre se muestra
Datos no sincronizados de DR	*n*KB	Se muestra cuando el nodo remoto no está disponible o no es coherente
Progreso de sincronización de DR	*n*%	Se visualiza cuando la sincronización está en curso
Hora estimada de finalización de DR	*AAAA-MM-DD HH:MM:SS*	Se visualiza cuando la sincronización está en curso
Progreso de la reversión de instantánea	*n*%	Se visualiza cuando el estado de DR es `Reverting to snapshot`. El estado se cuenta hacia atrás, así que 0% muestra que ha finalizado
Última sincronización de DR	*AAAA-MM-DD HH:MM:SS*	Muestra cuando los datos de DR están sin sincronizar (después de la sincronización inicial). Proporciona la hora y la fecha cuando los datos se han sincronizado por última vez.

Ejemplo

Ejemplo de estado normal en el nodo primario:

```
Queue manager status:                   Running
CPU:                                    0.00
Memory:                                 123MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
DR role:                                Primary
DR status:                              Normal
DR type:                                Synchronous
DR port:                                3000
DR local IP address:                    192.168.20.1
DR remote IP address:                   192.168.20.2
```

Ejemplo de estado normal en un nodo secundario:

```
Queue manager status:                   Ended immediately
DR role:                                Secondary
DR status:                              Normal
DR port:                                3000
DR local IP address:                    192.168.20.2
DR remote IP address:                   192.168.20.1
```

Ejemplo de estado en el nodo primario cuando la sincronización está en curso:

```
Queue manager status:                   Running
CPU:                                    0.53
Memory:                                 124MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
DR role:                                Primary
DR status:                              Synchronization in progress
DR type:                                Synchronous
DR port:                                3000
DR local IP address:                    192.168.20.1
DR remote IP address:                   192.168.20.2
DR synchronization progress:            11.0%
DR estimated time to completion:        2017-09-06 14:55:05
```

Ejemplo de un nodo primario, que muestra que está particionado:

```
Queue manager status:                   Running
CPU:                                    0.02
Memory:                                 124MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
DR role:                                Primary
DR status:                              Partitioned
DR type:                                Synchronous
DR port:                                3000
DR local IP address:                    192.168.20.1
DR remote IP address:                   192.168.20.2
```

Ejemplo de un nodo primario, que muestra que está sin sincronizar con el nodo secundario:

```
Queue manager status:                   Running
CPU:                                    0.00
Memory:                                 123MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
DR role:                                Primary
DR status:                              Remote unavailable
DR type:                                Asynchronous
DR port:                                3000
DR local IP address:                    192.168.20.1
DR remote IP address:                   192.168.20.2
DR out of sync data:                    15932KB
DR last in sync:                        2020-07-27 16:01:47
```

Ejemplo de un estado de resumen que muestra una discrepancia entre la versión del kernel del sistema operativo (RHEL 9.3) y el módulo de kernel DRBD (destinado a RHEL 9.2). Aunque el estado indica que se ha cargado el módulo de kernel de DRBD y se está ejecutando el gestor de colas esperado, en esta situación se debe actualizar el módulo de kernel de DRBD con la versión destinada al kernel del sistema operativo.

```
Node:                                   mqhavm07.exampleco.com
OS kernel version:                      5.14.0-362.18.1
DRBD OS kernel version:                 5.14.0-284.11.1
DRBD version:                           9.2.7+ptf.14
DRBD kernel module status:              Loaded

Queue manager name:                     DRQM8
Queue manager status:                   Ended immediately
DR role:                                Secondary

Queue manager name:                     DRQM7
Queue manager status:                   Running
DR role:                                Primary
```

Ejemplo de un estado de resumen que muestra una discrepancia entre la versión de kernel del sistema operativo (RHEL 8.10) y el módulo de kernel DRBD (destinado a RHEL 8.8). En este ejemplo, la discrepancia de versiones es más grave y el módulo de kernel de DRBD no se puede cargar correctamente. QM3 es un gestor de colas DR y está pensado para ser la instancia primaria, pero puesto que el módulo de kernel DRBD no se ha cargado completamente, se notifica como secundario con un estado de DR de Unknown. Para resolver esta anomalía, el módulo de kernel de DRBD debe actualizarse con la versión de destino para el kernel del sistema operativo en ejecución.

```
Node:                                   mqhavm57.exampleco.com
OS kernel version:                      4.18.0-553
DRBD OS kernel version:                 4.18.0-477
DRBD version:                           9.2.7+ptf.14
DRBD kernel module status:              Partially loaded

Queue manager name:                     QM3
Queue manager status:                   Status not available
DR role:                                Secondary
DR status:                              Unknown
```

Referencia relacionada

Linux rdqmstatus

Linux *Funcionamiento en un entorno de recuperación tras desastre*

Existen diversas situaciones en las que es posible que desee conmutar al gestor de colas secundario en una configuración de recuperación tras desastre.

Recuperación tras desastre

Tras la pérdida completa del gestor de colas primario en el sitio principal, inicia el gestor de colas secundario en el sitio de recuperación. Las aplicaciones se reconectan al gestor de colas en el sitio de recuperación y el gestor de colas secundario procesa los mensajes de la aplicación. Los pasos realizados para volver a la configuración anterior dependen de la causa de la anomalía. Por ejemplo, la pérdida completa del nodo principal frente a una pérdida temporal.

Para ver los pasos que se deben realizar tras una pérdida temporal del sitio principal, consulte “Conmutación a un nodo de recuperación” en la página 646. Para ver los pasos que se deben realizar tras una anomalía permanente, consulte “Sustitución de un nodo que ha fallado en una configuración de recuperación tras desastre” en la página 647.

Soporte de prueba de recuperación tras desastre

Puede probar la configuración de recuperación tras desastre conmutando temporalmente a la instancia secundaria y comprobando que las aplicaciones pueden conectarse satisfactoriamente. Sigue el mismo procedimiento que cuando conmuta tras una anomalía temporal del nodo primario, consulte “Conmutación a un nodo de recuperación” en la página 646.

Restaurando a la instantánea

Si sufre una anomalía en el nodo primario mientras hay una sincronización en curso, puede volver a la instantánea de los datos del gestor de colas secundario realizada justo antes del inicio de la sincronización. A continuación, el nodo secundario se restaura a un estado coherente y se puede ejecutar como primario. Para volver a la instantánea, convierta el secundario en el primario, tal como se describe en “Conmutación a un nodo de recuperación” en la página 646. Antes de iniciar el gestor

de colas, debe comprobar que se ha completado la reversión a la instantánea (mediante el mandato **rdqmstatus**).

Linux *Conmutación a un nodo de recuperación*

Si se produce un desastre en el sitio principal, realice los pasos para conmutar al sitio de recuperación.

Acerca de esta tarea

Tras la pérdida de un gestor de colas primario en el sitio principal, convierta el gestor de colas secundario en el sitio de recuperación en el gestor de colas primario e inícielo. Las aplicaciones se reconectan al gestor de colas en el sitio de recuperación y el gestor de colas procesa los mensajes de la aplicación. También puede utilizar este procedimiento para probar el nodo de recuperación.

Importante: Debe asegurarse de que la instancia primaria de un gestor de colas no puede ejecutarse o se ha detenido y se ha convertido en una instancia secundaria, antes de promover la instancia secundaria original. De lo contrario, se pueden acumular datos particionados.

Debe haber iniciado la sesión como root o haber iniciado la sesión como usuario que pertenece al grupo `mqm` y tiene la configuración sudo necesaria.

Procedimiento

1. Si utiliza este procedimiento para probar el gestor de colas secundario (es decir, la instancia primaria aún se está ejecutando), debe detener la instancia primaria y volver a asignarla como la instancia secundaria:

   ```
   endmqm qmname
   rdqmdr -m qmname -s
   ```

2. Convierta el gestor de colas secundario en el gestor de colas primario especificando el mandato siguiente en el nodo de recuperación:

   ```
   rdqmdr -m qmname -p
   ```

3. Inicie el gestor de colas especificando el mandato siguiente:

   ```
   strmqm qmname
   ```

4. Asegúrese de que las aplicaciones se vuelven a conectar al gestor de colas en el gestor de colas de recuperación. A menos que haya definido los canales con una lista de nombres de conexión alternativos, especificando los gestores de cola primario y secundario, las aplicaciones se conectarán automáticamente al nuevo gestor de colas primario.

Qué hacer a continuación

Cuando se restaura el nodo anómalo, siempre que el enlace entre los dos nodos esté funcionando, el gestor de colas no puede iniciarse en este nodo porque se está ejecutando en el nodo de recuperación donde ha promovido la instancia del gestor de colas secundario. Para volver al funcionamiento normal, debe detener el gestor de colas en el nodo de recuperación y, a continuación, promover el gestor de colas en el nodo original de nuevo al rol primario.

Referencia relacionada

strmqm

rdqmdr

Prueba del gestor de colas RDQM de recuperación

Puede probar que la instancia de recuperación de un gestor de colas en una configuración de recuperación tras desastre de RDQM funciona correctamente sin alterar el funcionamiento del sitio principal.

Acerca de esta tarea

Pruebe el gestor de colas de recuperación inhabilitando la interfaz entre los nodos principal y de recuperación. Convierte el gestor de colas secundario en el primario y, a continuación, puede probar el gestor de colas autónomo. Una vez finalizada la prueba, restaure la interfaz y suprima el gestor de colas de prueba. A continuación, vuelva a crear el gestor de colas como el gestor de colas secundario en la configuración de la recuperación tras desastre.

Procedimiento

1. Inhabilite la conexión de red entre el nodo principal y el nodo de recuperación.
2. En el nodo de recuperación, haga que el gestor de colas sea el primario:

   ```
   rdqmdr -m QMname -p
   ```

   Donde *nombreGC* es el nombre del gestor de colas.
3. Inicie el gestor de colas:

   ```
   strmqm QMname
   ```
4. Conecte las aplicaciones al gestor de colas y pruebe si funcionan según lo previsto.
5. Finalice el gestor de colas:

   ```
   endmqm QMname
   ```
6. Suprima el gestor de colas:

   ```
   dltmqm QMname
   ```
7. Restaure la conexión de red entre los dispositivos principal y de recuperación.
8. En el nodo principal, ejecute el mandato siguiente para recuperar el mandato **crtmqm** que ha utilizado al configurar por primera vez la recuperación tras desastre.

   ```
   rdqmdr -d -m QMname
   ```
9. Ejecute el mandato **crtmqm** resultante en el nodo de recuperación para volver a crear el gestor de colas secundario. El gestor de colas primario del nodo principal sincroniza sus datos con el gestor de colas secundario para actualizarlo.

Linux *Sustitución de un nodo que ha fallado en una configuración de recuperación tras desastre*

Si pierde uno de los nodos de una configuración de recuperación tras desastre, puede sustituir el nodo y restaurar la configuración de recuperación tras desastre mediante este procedimiento.

Acerca de esta tarea

Si se produce un desastre de modo que el nodo del sitio principal está más allá de la reparación, puede sustituir el nodo fallido, mientras se ejecuta el gestor de colas en el nodo de recuperación y, a continuación, se restaura la configuración de recuperación tras desastre original. El nodo de sustitución debe asumir la identidad del nodo que ha fallado: el nombre y la dirección IP deben coincidir.

Debe haber iniciado la sesión como root o haber iniciado la sesión como usuario que pertenece al grupo `mqm` y tiene la configuración sudo necesaria.

Procedimiento

Tras la pérdida del gestor de colas en el sitio principal, siga estos pasos:

1. En el nodo de recuperación, ejecute los mandatos siguientes para hacer que el gestor de colas secundario asuma la función principal:

   ```
   rdqmdr -m QMname -p
   ```

   Donde *nombreGC* es el nombre del gestor de colas.
2. Recupere el mandato que necesitará ejecutar en el nodo primario de sustitución para volver a configurar la recuperación tras desastre:

   ```
   rdqmdr -m QMname -d
   ```

   Copie la salida de este mandato.
3. Ejecute el mandato siguiente para iniciar el gestor de colas:

   ```
   strmqm QMname
   ```
4. Asegúrese de que las aplicaciones se vuelven a conectar al gestor de colas en el nodo de recuperación. A menos que haya definido los canales con una lista de nombres de conexión alternativos, especificando los gestores de cola primario y secundario, las aplicaciones se conectarán automáticamente al nuevo gestor de colas primario.
5. Sustituya el nodo que ha fallado en el sitio principal y configúrelo para que tenga el mismo nombre y la misma dirección IP que ha utilizado para la recuperación tras desastre en el nodo original. A continuación, configure la recuperación tras desastre ejecutando el mandato **crtmqm** que ha copiado en el paso 2. Ahora tiene una instancia secundaria del gestor de colas y la instancia primaria sincroniza sus datos con la instancia secundaria.
6. Finalice la instancia primaria actual.
7. Cuando se haya completado la sincronización, convierta la instancia primaria en ejecución en el nodo de recuperación de nuevo en la instancia secundaria:

   ```
   rdqmdr -m QMname -s
   ```
8. En el nodo primario de sustitución, convierta la instancia secundaria del gestor de colas en la instancia primaria:

   ```
   rdqmdr -m QMname -p
   ```
9. En el nodo primario de sustitución, inicie el gestor de colas:

   ```
   strmqm QMname
   ```

   Ahora ha restaurado la configuración tal como estaba antes de que se produjera la anomalía en el sitio principal.

Referencia relacionada

strmqm
rdqmdr
endmqm

Resolución de un problema incoherente en DR RDQM

Se puede informar de un estado de recuperación tras desastre de `inconsistent` si falla la sincronización entre las instancias primaria y secundaria de un gestor de colas.

Acerca de esta tarea

Se notifica un estado incoherente en la instancia secundaria de un gestor de colas porque se ha perdido la conexión de réplica con la instancia primaria durante una operación de sincronización. Es posible que tenga que realizar una acción para resolver esta situación. Considere la secuencia de sucesos siguiente:

1. Gestor de colas primario de DR en sincronización con el gestor de colas secundario de DR
2. Se ha perdido el enlace de réplica entre el primario y el secundario
3. Se ha restaurado un enlace de réplica entre el primario y el secundario
4. Se produce una resincronización en la que el gestor de colas secundario de DR se pone al día con el gestor de colas primario de DR. Durante este tiempo, se informa del estado de recuperación tras desastre de `synchronization in progress` para ambos gestores de colas.
5. Si la réplica se pierde de nuevo durante la resincronización, el estado en el secundario DR se notifica como `Inconsistent`.

Si el nodo que aloja el gestor de colas primario sigue operativo, y se puede restaurar el enlace de réplica, la resincronización se produce automáticamente. El estado incoherente se resuelve sin que se tome ninguna acción.

Si el nodo que aloja el gestor de colas primario deja de estar operativo, puede resolver el estado incoherente implementando una acción de revertir a la instantánea en el gestor de colas secundario. Esta operación revierte los datos al último estado correcto conocido.

Procedimiento

Para resolver un estado incoherente:

1. En el nodo de recuperación, convierta la instancia secundaria en la instancia primaria:

   ```
   rdqmdr -m qmname -p
   ```

   Se inicia la acción de revertir la operación de instantánea.
2. En el nodo de recuperación, compruebe el estado del gestor de colas para ver cuándo se ha completado la acción de revertir operación de instantánea:

   ```
   rdqmstatus -m qmname
   ```
3. Cuando el estado del gestor de colas es `Normal`, se inicia el gestor de colas:

   ```
   strmqm qmname
   ```

Resolución de un problema particionado (cerebro divido) en DR RDQM
Se puede producir un problema particionado si ambos gestores de colas de un par de recuperación tras desastre se ejecutan en el rol primario a la vez.

Acerca de esta tarea

Si ha promocionado la instancia secundaria de un gestor de colas en el nodo de recuperación mientras la instancia primaria original se ha seguido ejecutando en el nodo principal, efectivamente tiene dos versiones del mismo gestor de colas en ejecución, cada uno con su propia vista de los datos del gestor de colas. El estado de DR para el gestor de colas en cada nodo se notifica como `Partitioned`.

Debe decidir cuál de los dos gestores de colas tiene la vista de datos más correcta, y conservar dicho conjunto, mientras descarta el otro. Utilice el mandato **rdqmdr** para completar esta operación.

Hay dos procedimientos. El primero describe el mantenimiento de los datos del nodo principal, el segundo describe el mantenimiento de los datos del nodo de recuperación.

Procedimiento

- Para conservar los datos del gestor de colas en el nodo principal:

a) Asegúrese de que ambas instancias del gestor de colas están detenidas.

b) Especifique que el gestor de colas en el nodo de recuperación es el secundario:

```
rdqmdr -m qmname -s
```

c) Especifique que el gestor de colas en el nodo principal es el primario:

```
rdqmdr -m qmname -p
```

La sincronización empieza, con los datos del gestor de colas en el nodo principal que se está copiando en el nodo de recuperación.

d) Compruebe el estado de la sincronización:

```
rdqmstatus -m qmname
```

e) Cuando la sincronización se haya completado, inicie el gestor de colas en el nodo principal:

```
strmqm qmname
```

- Para conservar los datos del gestor de colas en el nodo de recuperación:

a) Asegúrese de que ambas instancias del gestor de colas están detenidas.

b) Especifique que el gestor de colas en el nodo principal es el secundario:

```
rdqmdr -m qmname -s
```

c) Especifique que el gestor de colas en el nodo de recuperación es el primario.

```
rdqmdr -m qmname -p
```

La sincronización empieza, con los datos del gestor de colas en el nodo de recuperación que se está copiando en el nodo principal.

d) Compruebe el estado de la sincronización:

```
rdqmstatus -m qmname
```

e) Cuando la sincronización se haya completado, degrade el gestor de colas en el nodo de recuperación:

```
rdqmdr -m qmname -s
```

f) Promocione el gestor de colas en el nodo principal e inícielo:

```
rdqmdr -m qmname -p
strmqm qmname
```

Modificación de las direcciones IP en configuraciones de recuperación tras desastre
Si cambia las direcciones IP de cualquiera de las interfaces en una configuración de recuperación tras desastre, la réplica ya no es posible entre los dos nodos.

Si necesita cambiar direcciones IP para la interfaz de réplica para cualquiera de los nodos DR, debe utilizar el procedimiento siguiente:

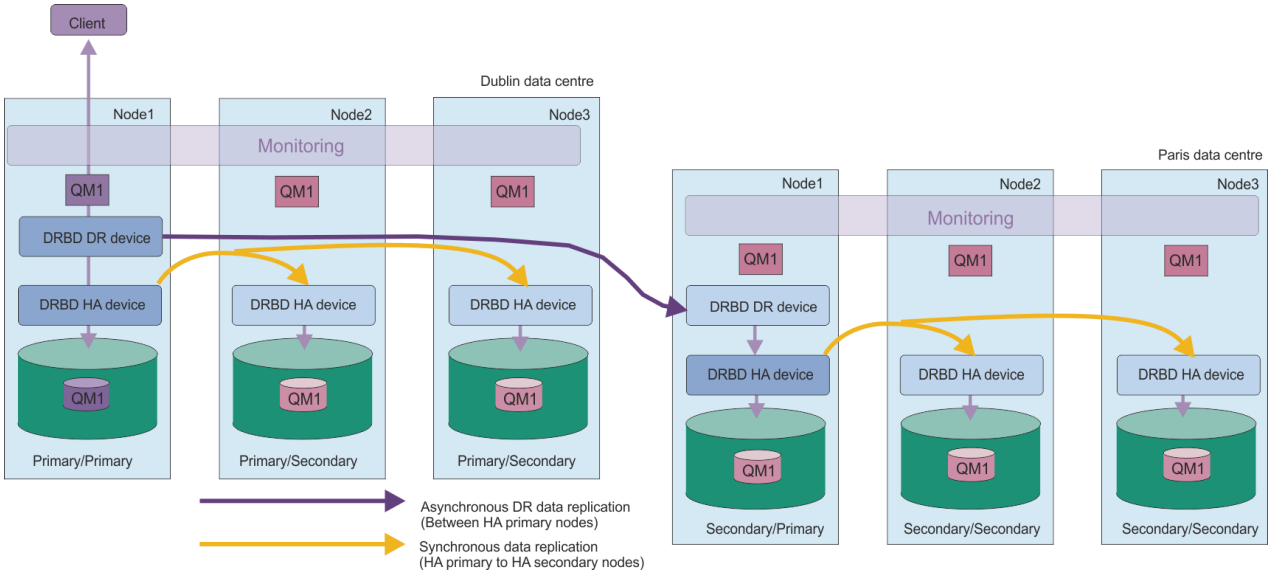
1. En el nodo primario, realice una copia de seguridad de los gestores de colas DR y, a continuación, suprímalos. En el nodo de recuperación, suprima los gestores de colas. Consulte “Copia de seguridad y restauración de datos de gestor de colas de IBM MQ” en la página 706 y “Supresión de un RDQM de DR” en la página 633.
2. Vuelva a crear los gestores de colas DR, especificando las nuevas direcciones IP y restaure las copias de seguridad, consulte “Creación de un RDQM de recuperación tras desastre” en la página 630 y “Copia de seguridad y restauración de datos de gestor de colas de IBM MQ” en la página 706.

Linux Recuperación tras desastre y alta disponibilidad de RDQM

Puede configurar un gestor de colas de datos replicados (RDQM) que se ejecute en un grupo de alta disponibilidad en un sitio, pero que pueda migrar tras error a otro grupo de alta disponibilidad en otro sitio si se produce algún desastre que hace que el primer grupo no esté disponible. Esto se conoce como RDQM de DR/HA.

Un RDQM de DR/HA combina las características de un RDQM de alta disponibilidad (consulte “Alta disponibilidad en RDQM” en la página 595) y de un RDQM de recuperación tras desastre (consulte “Recuperación tras desastre de RDQM” en la página 626).

El diagrama siguiente muestra un ejemplo de DR/HA RDQM.



La réplica entre los RDQM de DR/HA en el sitio principal y el sitio de recuperación tras desastre siempre es asíncrona. Con la réplica asíncrona, las operaciones como PUT o GET de IBM MQ se completan y devuelven a la aplicación antes de que el suceso se duplique en el gestor de colas secundario.

Puede tener dos sitios activos en lugar de los sitios 'main' y 'recovery', si es necesario, por lo que algunos de los RDQM DR/HA se ejecutan en un sitio y algunos en el otro durante la operación normal. Si se produce un desastre y un sitio deja de estar disponible, todos los RDQM de DR/HA se ejecutan en el mismo grupo HA y en el mismo sitio.

Cada grupo HA se configura de la misma forma que un grupo HA ordinario. Puede definir direcciones IP flotantes para un RDQM de DR/HA en cada grupo HA. La dirección IP flotante puede ser la misma o distinta para cada grupo HA.

No puede actualizar un RDQM existente para que sea un RDQM de DR/HA, debe crear un RDQM de DR/HA. (Si es necesario, puede hacer una copia de seguridad de los datos de un RDQM existente, suprimirlo, volver a crearlo como RDQM de DR/HA y, a continuación, restaurar los datos, consulte “Copia de seguridad y restauración de datos de gestor de colas de IBM MQ” en la página 706).

Para configurar los RDQM de DR/HA, debe completar los pasos principales siguientes:

1. Configure un grupo HA en el sitio 'main'.
2. Configure un grupo HA en el sitio 'recovery'.
3. Cree un DR/HA primario/primario DR/HA primario en un nodo del grupo HA en el sitio 'main'.
4. Cree RDQM de DR/HA primarios/secundarios en los otros dos nodos en el sitio 'main'.
5. Defina una dirección IP flotante para que una aplicación acceda al RDQM de DR/HA cuando se ejecute en cualquiera de los nodos del grupo HA en el sitio 'main'.
6. Cree un RDQM de DR/HA secundario/primario en un nodo del grupo HA en el sitio 'recovery'.
7. Cree RDQM de DR/HA secundarios/secundarios en los otros dos nodos en el sitio 'recovery'.

8. Defina una dirección IP flotante para que una aplicación acceda al RDQM de DR/HA cuando se ejecute en cualquiera de los nodos del grupo HA en el sitio 'recovery'.

Los detalles sobre cada uno de estos pasos se proporcionan en los temas siguientes.

Linux *Requisitos para una solución RDQM de DR/HA*

Los requisitos para la solución RDQM de DR/HA son los mismos que para la solución RDQM de HA y la solución RDQM de DR.

Para obtener detalles sobre los requisitos de las partes de HA de la configuración, consulte “Requisitos de la solución de HA de RDQM” en la página 598.

Para obtener detalles sobre la parte de DR de la configuración, consulte “Requisitos de la solución de RDQM de DR” en la página 628.

Linux *Configuración de grupos HA para RDQM DR/HA*

Debe crear un grupo HA en los sitios principales y de recuperación. Si tiene un grupo HA existente en cualquiera de los dos sitios, puede crear RDQM DR/HA en ese grupo HA. (Los RDQM existentes seguirán funcionando como antes).

El procedimiento es el mismo que el descrito para la alta disponibilidad de RDQM, consulte “Definición del clúster de Pacemaker (grupo HA)” en la página 601.

Al definir un grupo de alta disponibilidad, especifique las direcciones IP utilizadas para la supervisión y la duplicación por cada nodo del archivo `rdqm.ini`. Al crear un grupo HA para admitir los RDQM DR/HA también puede especificar las direcciones IP utilizadas para la réplica de DR por el grupo HA que va a definir y las direcciones IP utilizadas para la réplica de DR por los nodos del otro grupo HA del par de DR. (Si no especifica las direcciones IP de duplicación DR en el archivo `rdqm.ini`, puede especificarlas en la línea de mandatos cuando cree un DR/HA RDQM.)

Si está configurando un grupo de HA existente, puede añadir direcciones IP de duplicación DR al archivo `rdqm.ini` existente. No es necesario que vuelva a ejecutar **rdqmadm** después de actualizar `rdqm.ini`, pero debe actualizar `rdqm.ini` antes de crear los RDQM DR/HA.

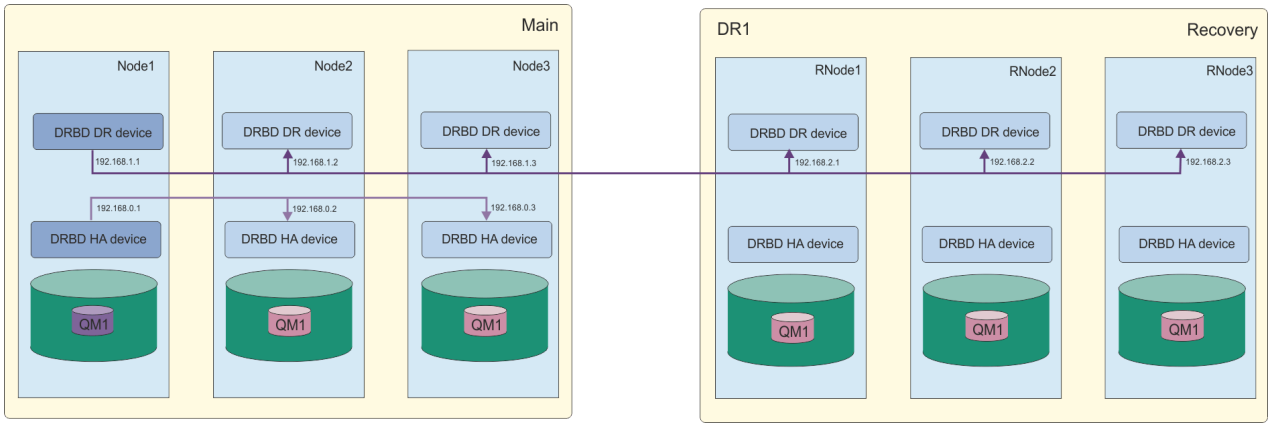
Utilice el atributo `DR_Replication` en las stanzas `Node` para especificar las interfaces de réplica de DR en el grupo HA que está definiendo, por ejemplo:

```
Node:
  Name=Node1
  HA_Replication=192.168.0.1
  DR_Replication=192.168.1.1
Node:
  Name=Node2
  HA_Replication=192.168.0.2
  DR_Replication=192.168.1.2
Node:
  Name=Node3
  HA_Replication=192.168.0.3
  DR_Replication=192.168.1.3
```

Utilice la stanza `DRGroup` para especificar las direcciones de réplica de DR del grupo HA remoto, por ejemplo:

```
DRGroup:
  Name=DR1
  DR_Replication=192.168.2.1
  DR_Replication=192.168.2.2
  DR_Replication=192.168.2.3
```

El diagrama siguiente ilustra esta configuración:



Si no especifica las direcciones IP de réplica de DR para los nodos del grupo HA local en el archivo `rdqm.ini` o en la línea de mandatos cuando cree un RDQM de DR/HA, se utilizarán las interfaces `HA_Replication` definidas para cada nodo para la réplica de DR. Debe especificar las direcciones de réplica de DR del grupo HA remoto en el archivo `rdqm.ini` o en la línea de mandatos **`crtmqm`**.

Linux *Creación de RDQM de DR/HA*

Puede utilizar el mandato **`crtmqm`** para crear un gestor de colas de datos replicados (RDQM) en una configuración DR/HA.

Acerca de esta tarea

Puede crear un RDQM de DR/HA como un usuario en el grupo `mqm` si el usuario puede utilizar sudo. De lo contrario, debe crear el RDQM como root.

Debe crear un número de RDQM de DR/HA:

- En el grupo HA en el sitio 'main':
  - En el nodo donde desea que se ejecute el gestor de colas en condiciones normales, cree el RDQM de DR/HA primario/secundario.
  - En cada uno de los otros dos nodos del grupo HA, cree un RDQM de DR/HA primario/secundario.
- En el grupo HA en el sitio 'recovery':
  - En el nodo en el que se ejecutará el gestor de colas si realiza una migración tras error al sitio de recuperación, cree el RDQM de DR/HA secundario/primario. Puede utilizar la salida del mandato cuando creó el gestor de colas primario/primario en el sitio 'main'.
  - En cada uno de los otros dos nodos del grupo HA, cree un RDQM de DR/HA secundario/secundario.

Todas las instancias de gestor de colas deben tener el mismo nombre y deben tener asignada la misma cantidad de almacenamiento.

Los puntos siguientes proporcionan algunas directrices sobre el dimensionamiento del sistema de archivos del gestor de colas:

1. Al crear un gestor de colas RDQM, se asigna un sistema de archivos para almacenar datos y registros del gestor de colas. Es importante dimensionar este sistema de archivos de forma adecuada para que el gestor de colas pueda registrar la actividad en curso en sus registros y almacenar los mensajes de aplicación en las colas. Al dimensionar el sistema de archivos, tenga en cuenta los requisitos máximos de mensajería, el crecimiento futuro de la carga de trabajo y las paradas de las aplicaciones que pueden hacer que los mensajes se acumulen en las colas. Para obtener instrucciones sobre cómo calcular el tamaño del registro de recuperación del gestor de colas, consulte “¿Qué tamaño debe tener el sistema de archivos de registro?” en la página 685. Al calcular los requisitos de almacenamiento para los mensajes de aplicación, es necesario tener en cuenta el tamaño y el número de mensajes, además de su cabecera MQMD y cualquier propiedad de mensaje que tengan.

2. Los sistemas de archivos del gestor de colas RDQM no se pueden redimensionar dinámicamente. Debe realizar una copia de seguridad y, a continuación, restaurar un gestor de colas RDQM con un sistema de archivos más grande si es necesario, consulte "Redimensionar el sistema de archivos para un gestor de colas RDQM HA" en la página 611.
3. Puede limitar el tamaño de colas individuales en disco utilizando atributos de cola local, como MAXDEPTH y MAXFSIZE. Consulte Modificación de archivos de cola de IBM MQ.
4. Debe supervisar el uso de disco en curso y responder adecuadamente si el uso de disco aumenta antes de que el uso del sistema de archivos pase a ser crítico. El uso del sistema de archivos se puede supervisar utilizando las prestaciones de plataforma/sistema operativo o suscribiéndose a las métricas publicadas en los temas del sistema IBM MQ que se describen en Métricas publicadas en los temas del sistema.

Procedimiento

- Para crear el RDQM de DR/HA primario/primario:
  a) Escriba el mandato siguiente:

```
crtmqm -sx -rr p
              [-rl DRLocalIP1,DRLocalIP2,DRLocalIP3]
              (-ri DRRemoteIP1,DRRemoteIP2,DRRemoteIP3 | -rn GroupName)
              -rp DRPort
              [-z] [-q] [-c Text] [-d DefXmitQ] [-h MaxHandles]
              [-g ApplicationGroup] [-oa user|group]
              [-t TrigInt] [-u DeadQ] [-x MaxUMsgs]
              [-lp LogPri] [-ls LogSec]
              [-lc | -ll | -lla | -lln] [-lf LogFileSize]
              [-p Port] [-fs FilesystemSize] QMgrName
```

  Donde:

  -sx
  Indica que el rol de HA inicial es primario.

  -rr p
  Indica que el rol de DR inicial es primario.

  -rl *IPLocalDR1*, *IPLocalDR2*, *IPLocalDR3*
  De forma opcional, especifique las direcciones IP de las interfaces DR en los tres nodos en el sitio local (es decir, el sitio 'main'). Si no se especifica, se utilizan las direcciones IP especificadas en el archivo `rdqm.ini`.

  -ri *IPRemotaDR1*, *IPRemotaDR2*, *IPRemotaDR3*
  Especifique las direcciones IP de las interfaces DR en los tres nodos en el sitio remoto (es decir, el sitio 'recovery'). Debe especificar este parámetro o el parámetro `-rn`.

  -rn *NombreGrupo*
  Especifique el nombre de grupo de HA remoto tal como se especifica en el archivo `rdqm.ini`. Debe especificar `-ri` o `-rn`.

  -rp *Puerto*
  Especifica el puerto que se utilizará para la réplica de DR.

  otras_opciones_crtmqm
  Puede especificar opcionalmente una o varias de estas opciones generales de **crtmqm**:

  - -z
  - -q
  - -c *Texto*
  - -d *ColaTransmisiónPredeterminada*
  - -h *MaxManejadores*
  - -g *GrupoAplicaciones*
  - -oa user | group

- -t *TrigInt*
- -u *ColaMsjNoEntregados*
- -x *MaxMsjU*
- -lp *RegPri*
- -ls *RegSec*
- -lc | -l
- -lla | -lln
- -lf *TamañoArchivoRegistro*
- -p *Puerto*

-fs *tamaño*
De forma opcional, especifica el tamaño del sistema de archivos para crear el gestor de colas, es decir, el tamaño del volumen lógico que se crea en el grupo de volúmenes drbdpool. También se crea otro volumen lógico de ese tamaño, para permitir la reversión a la operación de instantánea, de manera que el almacenamiento total del RDQM de DR es apenas el doble que el especificado aquí.

Tamaño es un valor numérico, que se especifica en GB. Puede especificar un valor en MB especificando el valor seguido del carácter M. Por ejemplo, para especificar un tamaño de sistema de archivos de 3 GB, especifique 3. Para especificar un tamaño de sistema de archivos de 1024 MB, especifique 1024M. (También puede añadir un sufijo G para indicar explícitamente GB.)

nombreGC
Especifica el nombre del gestor de colas de datos replicados. El nombre es sensible a las mayúsculas y minúsculas.

Una vez completado el mandato, genera el mandato que puede especificar en el sitio de recuperación para crear la instancia secundaria/primaria del gestor de colas.

- Para crear un RDQM de DR/HA primario/secundario en los otros dos nodos del grupo HA:

  a) Especifique el mandato siguiente en cada nodo:

```
crtmqm -sxs -rr p
              [-rl DRLocalIP1,DRLocalIP2,DRLocalIP3]
              (-ri DRRemoteIP1,DRRemoteIP2,DRRemoteIP3 | -rn GroupName)
              -rp DRPort
              [-fs FilesystemSize] QMgrName
```

Donde:

-sxs
Indica que el rol de HA inicial es secundario.

-rr p
Indica que el rol de DR inicial es primario.

-rl *IPLocalDR1*, *IPLocalDR2*, *IPLocalDR3*
De forma opcional, especifique las direcciones IP de las interfaces DR en los tres nodos en el sitio local (es decir, el sitio 'main'). Si no se especifica, se utilizan las direcciones IP especificadas en el archivo `rdqm.ini`.

-ri *IPRemotaDR1*, *IPRemotaDR2*, *IPRemotaDR3*
Especifique las direcciones IP de las interfaces DR en los tres nodos en el sitio remoto (es decir, el sitio 'recovery'). Debe especificar este parámetro o el parámetro `-rn`.

-rn *NombreGrupo*
Especifique el nombre de grupo de HA remoto tal como se especifica en el archivo `rdqm.ini`. Debe especificar `-ri` o `-rn`.

-rp *Puerto*
Especifica el puerto que se utilizará para la réplica de DR.

-fs *tamaño*
Especifica el tamaño del sistema de archivos que se debe crear para el gestor de colas, es decir, el tamaño del volumen lógico que se crea en el grupo de volúmenes drbdpool. Si ha especificado un tamaño no predeterminado al crear el RDQM primario/primario, debe especificar el mismo valor aquí.

Tamaño es un valor numérico, que se especifica en GB. Puede especificar un valor en MB especificando el valor seguido del carácter `M`. Por ejemplo, para especificar un tamaño de sistema de archivos de 3 GB, especifique 3. Para especificar un tamaño de sistema de archivos de 1024 MB, especifique 1024M. (También puede añadir un sufijo G para indicar explícitamente GB.)

nombreGC
Especifica el nombre del RDQM primario/secundario. Debe ser el mismo que el nombre especificado para la instancia primaria/primaria del RDQM. Tenga en cuenta que el nombre es sensible a mayúsculas y minúsculas.

- Para crear un RDQM de DR/HA secundario/primario en el nodo en el que se ejecutará el gestor de colas si éste realiza la migración tras error al sitio de recuperación:

  a) Utilice la salida del mandato cuando haya creado el DR/HA primario/primario en el sitio principal o especifique el mandato siguiente:

```
crtmqm -sx -rr s
             [-rl DRLocalIP1,DRLocalIP2,DRLocalIP3]
             (-ri DRRemoteIP1,DRRemoteIP2,DRRemoteIP3 | -rn GroupName)
             -rp DRPort
             [-fs FilesystemSize] QMgrName
```

  -sx
  Indica que el rol de HA inicial es primario.

  -rr s
  Indica que el rol de DR inicial es secundario.

  -rl *IPLocalDR1*, *IPLocalDR2*, *IPLocalDR3*
  De forma opcional, especifique las direcciones IP de las interfaces DR en los tres nodos en el sitio local (es decir, el sitio 'recovery'). Si no se especifica, se utilizan las direcciones IP especificadas en el archivo `rdqm.ini`.

  -ri *IPRemotaDR1*, *IPRemotaDR2*, *IPRemotaDR3*
  Especifique las direcciones IP de las interfaces DR en los tres nodos en el sitio remoto (es decir, el sitio 'main'). Debe especificar este parámetro o el parámetro `-rn`.

  -rn *NombreGrupo*
  Especifique el nombre de grupo de HA remoto tal como se especifica en el archivo `rdqm.ini`. Debe especificar `-ri` o `-rn`.

  -rp *Puerto*
  Especifica el puerto que se utilizará para la réplica de DR.

  -fs *tamaño*
  De forma opcional, especifica el tamaño del sistema de archivos para crear el gestor de colas, es decir, el tamaño del volumen lógico que se crea en el grupo de volúmenes drbdpool. También se crea otro volumen lógico de ese tamaño, para permitir la reversión a la operación de instantánea, de manera que el almacenamiento total del RDQM de DR es apenas el doble que el especificado aquí.

  nombreGC
  Especifica el nombre del gestor de colas de datos replicados. El nombre es sensible a las mayúsculas y minúsculas.

- Para crear un RDQM de HA/DR secundario/secundario en los otros dos nodos en el sitio de recuperación:

  a) Especifique el mandato siguiente en cada nodo:

```
crtmqm -sxs -rr s
             [-rl DRLocalIP1,DRLocalIP2,DRLocalIP3]
```

```
(-ri DRRemoteIP1,DRRemoteIP2,DRRemoteIP3 | -rn GroupName)
-rp DRPort
[-fs FilesystemSize] QMgrName
```

-sxs
Indica que el rol de HA inicial es primario.

-rr s
Indica que el rol de DR inicial es secundario.

-rl *IPLocalDR1*, *IPLocalDR2*, *IPLocalDR3*
De forma opcional, especifique las direcciones IP de las interfaces DR en los tres nodos en el sitio local. Si no se especifica, se utilizan las direcciones IP especificadas en el archivo `rdqm.ini`.

-ri *IPRemotaDR1*, *IPRemotaDR2*, *IPRemotaDR3*
Especifique las direcciones IP de las interfaces DR en los tres nodos en el sitio remoto. Debe especificar este parámetro o el parámetro `-rn`.

-rn *NombreGrupo*
Especifique el nombre de grupo de HA remoto tal como se especifica en el archivo `rdqm.ini`. Debe especificar `-ri` o `-rn`.

-rp *Puerto*
Especifica el puerto que se utilizará para la réplica de DR.

-fs *tamaño*
De forma opcional, especifica el tamaño del sistema de archivos para crear el gestor de colas, es decir, el tamaño del volumen lógico que se crea en el grupo de volúmenes drbdpool. También se crea otro volumen lógico de ese tamaño, para permitir la reversión a la operación de instantánea, de manera que el almacenamiento total del RDQM de DR es apenas el doble que el especificado aquí.

nombreGC
Especifica el nombre del gestor de colas de datos replicados. El nombre es sensible a las mayúsculas y minúsculas.

Nota: Cuando se crea un RDQM, se asigna el siguiente número de puerto libre por encima de 7000 para el enlace de réplica de HA. Si se descubre que el puerto elegido es utilizado por otra aplicación, el mandato **crtmqm** falla con el error AMQ6543 y dicho puerto se añade a una lista de exclusión. Debe suprimir las instancias secundarias del gestor de colas y, a continuación, ejecutar de nuevo el mandato **crtmqm**.

Qué hacer a continuación

Después de haber creado todos los RDQM de DR/HA, debe comprobar el estado de las instancias primaria/primaria y secundaria/primaria para comprobar que todas son correctas. Utilice el mandato **rdqmstatus** en los nodos. Los nodos deben mostrar el estado normal, tal como se describe en "Visualización del grupo RDQM de DR/HA y del estado del grupo HA" en la página 659. Si no muestran ese estado, suprima la instancia secundaria/primaria y vuelva a crearla, prestando atención a utilizar los argumentos correctos.

Tareas relacionadas

"Creación de RDQM de DR/HA" en la página 653
Puede utilizar el mandato **crtmqm** para crear un gestor de colas de datos replicados (RDQM) en una configuración DR/HA.

Referencia relacionada

crtmqm

Linux *Supresión de un RDQM DR/HA*

Puede utilizar el mandato **dltmqm** para suprimir un gestor de colas de datos replicados (RDQM) de DR/HA.

Acerca de esta tarea

Debe ejecutar el mandato para suprimir el RDQM en ambos nodos primario/primario y secundario/primario. Antes hay que parar el RDQM. El mandato se puede ejecutar como usuario mqm si dicho usuario tiene los privilegios sudo necesarios. De lo contrario, hay que ejecutar el mandato como root.

Procedimiento

- Para suprimir un RDQM de DR/HA, especifique el mandato siguiente:

```
dltmqm RDQM_name
```

Referencia relacionada

dltmqm

Linux *Creación de una dirección IP flotante*

Puede crear direcciones IP flotantes para cada uno de los grupos HA en una configuración de RDQM de DR/HA.

Una dirección IP flotante permite que un cliente utilice la misma dirección IP para un RDQM de DR/HA independientemente del nodo del grupo HA en que esté se ejecutando. Si los dos grupos HA tienen redes privadas o aisladas para la conectividad de aplicaciones, se puede definir la misma dirección IP flotante para ambos grupos. Sin embargo, todavía debe definir esa dirección IP flotante dos veces, una vez en cada uno de los grupos HA.

Puede crear y suprimir direcciones IP flotantes utilizando el mismo método que para un RDQM de HA. Consulte “Creación y borrado de una dirección IP flotante” en la página 614.

Linux *Inicio, detención y visualización del estado de un RDQM de DR/HA*

Puede utilizar variantes de mandatos de control de IBM MQ estándar para iniciar, detener y ver el estado actual de un RDQM de DR/HA.

Acerca de esta tarea

Debe ejecutar los mandatos que inician, detienen y visualizan el estado actual de un RDQM de DR/HA con un usuario que pertenezca a los grupos `mqm` y `haclient`.

Debe ejecutar los mandatos para iniciar y detener un gestor de colas en el nodo primario para ese gestor de colas.

Procedimiento

- Para iniciar un RDQM, ejecute el mandato siguiente en el nodo primario del RDQM:

```
strmqm qmname
```

  donde *nombreGC* es el nombre del RDQM de DR/HA que se quiere iniciar.

  El RDQM se inicia y Pacemaker empieza a gestionar el RDQM. Hay que especificar la opción `-ns` con `strmqm` si se desea especificar cualquier otra opción `strmqm`.
- Para detener un RDQM, ejecute el mandato siguiente en el nodo primario del RDQM de DR/HA:

```
endmqm qmname
```

  donde *nombreGC* es el nombre del RDQM que se desea parar.

  Pacemaker deja de gestionar el RDQM y este se termina. Todos los demás parámetros **endmqm** se pueden utilizar cuando se para un RDQM.
- Para ver el estado un RDQM, ejecute el mandato siguiente:

```
dspmq -m QMname
```

La información de estado que aparece en la salida depende de si se ejecuta el mandato en el nodo primario o secundario del RDQM. Si se ejecuta en el nodo primario, se mostrará uno de los mensajes de estado normal devueltos por **dspmq**. Si ejecuta el mandato en un nodo secundario, se muestra el estado `Ended immediately` . Por ejemplo, si se ejecuta **dspmq** en el nodo RDQM7, podría devolverse la siguiente información:

```
QMNAME(DRQM8)                           STATUS(Ended immediately)
QMNAME(DRQM7)                           STATUS(Running)
```

Puede utilizar argumentos con `dspmq` para establecer si un RDQM está configurado para la recuperación tras desastre, y si es actualmente la instancia primaria o secundaria:

```
dspmq -m QMname -o (dr | DR)
```

Se visualiza una de las respuestas siguientes:

DRROLE()
Indica que el gestor de colas no está configurado para la recuperación tras desastre.

DRROLE(Primary)
Indica que el gestor de colas está configurado como primario de DR.

DRROLE(Secondary)
Indica que el gestor de colas está configurado como secundario de DR.

Utilice el mandato **dspmq -o all** para ver la recuperación tras desastre y la información de alta disponibilidad de los RDQM de DR/HA. Por ejemplo, si ejecuta **dspmq -o all** en el nodo en el que se ejecuta el RDQM de DR/HA, verá la información de estado siguiente:

```
QMNAME(TESTQM1)                                           STATUS(Running) HA(Replicated)
DRROLE(Primary)
```

Referencia relacionada
dspmq (visualizar gestores de colas)
endmqm (finalizar gestor de colas)
strmqm (iniciar gestor de colas)

Acciones de recurso fallido en configuraciones de DR/HA

Las acciones de recurso fallido surgen cuando el componente Pacemaker de una configuración de alta disponibilidad en RDQM encuentra algún problema con un recurso en uno de los nodos de un grupo HA.

Las acciones de recurso fallido puede aparecer en cualquiera de las configuraciones de HA de una configuración de DR/HA de RDQM. Puede utilizar el mandato **rdqmstatus** para ver las acciones de recurso fallido y el mandato **rdqmclean** para borrarlas (una vez resuelta la causa de la anomalía). El proceso es el mismo que en las configuraciones de HA de RDQM sin el componente DR. Consulte “Acciones de recurso fallido” en la página 617 para obtener más información

Tareas relacionadas
“Visualización del grupo RDQM de DR/HA y del estado del grupo HA” en la página 659
Puede ver el estado de HA y el rol de DR de los gestores de colas de datos replicados (RDQM) de DR/HA.

“Visualización del estado de un RDQM y de un grupo HA” en la página 618
Se puede ver el estado de un grupo HA y de gestores de colas de datos replicados (RDQM) individuales.

Referencia relacionada
rdqmclean
rdqmstatus

Linux *Visualización del grupo RDQM de DR/HA y del estado del grupo HA*

Puede ver el estado de HA y el rol de DR de los gestores de colas de datos replicados (RDQM) de DR/HA.

Acerca de esta tarea

Puede utilizar el mandato **rdqmstatus** para ver el estado de los RDQM individuales u obtener una visión general del estado de todos los RDQM conocidos para el grupo HA.

El estado de resumen de un nodo también muestra información sobre el módulo de kernel de DRBD en el que se basa RDQM. Cuando se actualiza RDQM, es importante asegurarse de que se haya instalado la versión correcta del módulo de kernel de DRBD para la versión del kernel RHEL que se ejecuta en el sistema. El estado muestra la versión del kernel de sistema operativo, la versión del kernel para la que se ha creado el módulo DRBD, la versión de DRBD y el estado cargado del módulo de kernel de DRBD.

Nota: Tenga en cuenta que, en una configuración de HA/DR, la configuración de DR siempre utiliza la réplica asíncrona, mientras que la configuración de HA siempre utiliza la réplica síncrona. Estos valores no se visualizan en la salida del mandato `rdqmstatus -m qmgr` en una configuración HA/DR combinada.

Debe ser un usuario de los grupos `mqm` y `haclient` para ejecutar el mandato **rdqmstatus**. Puede ejecutar el mandato en cualquiera de los nodos de los grupos HA.

Procedimiento

- Para ver el estado de resumen de un nodo y los RDQM de forman parte de la configuración HA:

  ```
  rdqmstatus
  ```

  Se muestra la identidad del nodo en el que se ha ejecutado el mandato y el estado de los RDQM en la configuración de HA, más su rol de DR actual, por ejemplo:

  ```
  Node:                                 main-alice
  OS kernel version:                    5.14.0-362.18.1
  DRBD OS kernel version:               5.14.0-362.18.1
  DRBD version:                         9.2.7
  DRBD kernel module status:            Loaded

  Queue manager name:                   RDQM1
  Queue manager status:                 Running elsewhere
  HA current location:                  main-charlie
  HA preferred location:                main-charlie
  HA blocked location:                  None

  Queue manager name:                   RDQM9
  Queue manager status:                 Running elsewhere
  HA current location:                  main-bob
  HA preferred location:                main-bob
  HA blocked location:                  None
  DR role:                              Primary

  Queue manager name:                   RDQM7
  Queue manager status:                 Running
  HA current location:                  This node
  HA preferred location:                This node
  HA blocked location:                  None
  DR role:                              Primary
  ```

  En este ejemplo, RDQM7 y RDQM8 son los RDQM de DR/HA, mientras que RDQM1 es un RDQM de HA, que no está configurado para poder conmutar a un sitio de recuperación tras desastre.

  El estado del módulo de kernel de DRBD tiene uno de los siguientes valores:

  Cargado
  Indica que se ha cargado el módulo DRBD.

  Cargado parcialmente
  Puede producirse cuando se ha cargado el módulo DRBD, pero no funciona correctamente debido a una discrepancia.

  No cargado
  No se ha cargado el módulo DRBD. Esto se puede visualizar en una configuración recién instalada, cuando todavía no se ha creado ningún gestor de colas RDQM.

No instalado
Indica que el módulo DRBD no está instalado. o que IBM MQ no ha podido determinar la versión de kernel del sistema operativo del módulo DRBD.

Versión instalada anteriormente todavía cargada
Este estado puede producirse si se instala un nuevo módulo DRBD mientras se ejecuta el módulo DRBD existente (es decir, se está ejecutando un gestor de colas RDQM). El módulo recién instalado se notifica en el estado, pero no es el módulo que se está ejecutando realmente.

- Para ver el estado de un gestor de colas determinado en todos los nodos del grupo HA, ejecute el mandato siguiente:

```
rdqmstatus -m qmname
```

donde *nombreGC* es el nombre del RDQM cuyo estado se desea visualizar. Se muestra el estado del RDQM del nodo actual, seguido de un resumen del estado de los otros dos nodos desde la perspectiva del nodo actual.

- Para ver el estado de un gestor de colas determinado en todos los nodos del grupo HA, incluyendo los detalles de posibles acciones de recurso fallido, ejecute el mandato siguiente:

```
rdqmstatus -m qmname -a
```

donde *nombreGC* es el nombre del RDQM cuyo estado se desea visualizar. Se muestra el estado del RDQM del nodo actual, seguido de un resumen del estado de los otros dos nodos desde la perspectiva del nodo actual. Esto va acompañado de los detalles de cualquier posible acción de recurso fallido asociada con el RDQM.

La tabla siguiente resume la información sobre el nodo actual que puede devolver el mandato `rdqmstatus -m qmname` para un RDQM.

Tabla 36. Estado de nodo actual

Atributo de estado	Valores posibles	Cuándo se muestra
Nombre de nodo	*nombrenodo*	Siempre se muestra
Estado del gestor de colas	Estado del gestor de colas (uno de los estados que son válidos para el mandato **dspmq**)	Siempre se muestra
CPU	*n.nn*%	Solo se muestra cuando RDQM se ejecuta en este nodo
Memoria	*nnn* MB utilizados	Solo se muestra cuando RDQM se ejecuta en este nodo
Sistema de archivos del gestor de colas	*nnn*MB usadas, *y.y*GB asignadas [*z*%]	Solo se muestra cuando RDQM se ejecuta en este nodo
Rol de HA	Primario Secundario Desconocido	Siempre se muestra
Estado de HA	Todos los nodos en espera Este nodo en espera Nodos remotos en espera Mixto	Todos los nodos en espera Nodo actual en espera Ambos nodos remotos en espera Distintos estados por cada nodo remoto

Tabla 36. Estado de nodo actual (continuación)		
Atributo de estado	**Valores posibles**	**Cuándo se muestra**
Control de HA	Habilitada Inhabilitado Desconocido	Siempre se muestra. Indica si RDQM está bajo el control de Pacemaker
Ubicación de HA preferida	Ninguna Este nodo Desconocido *nombrenodo*	Siempre se muestra
Ubicación bloqueada de HA	Ninguna - El gestor de colas no está bloqueado y puede ejecutarse en cualquier nodo Este nodo - El gestor de colas está bloqueado y no puede ejecutarse en el nodo actual debido a una o más acciones de recurso fallido *nombrenodo* - El gestor de colas está bloqueado y no puede ejecutarse en el nodo *nombrenodo* debido a una o más acciones de recurso fallido *nombrenodo1*, *nombrenodo2* - El gestor de colas está bloqueado y no puede ejecutarse en *nombrenodo1* y *nombrenodo2* debido a una o más acciones de recurso fallido Todos los nodos - El gestor de colas está bloqueado y no puede ejecutarse en ningún nodo debido a una o más acciones de recurso fallido	Siempre se muestra
Interfaz de IP flotante de HA	*nombre_interfaz*	Siempre se muestra
Dirección IP flotante de HA	*IPV4_address*	Siempre se muestra
Rol de DR	Primario Secundario Secundario pendiente Desconocido	Siempre se muestra

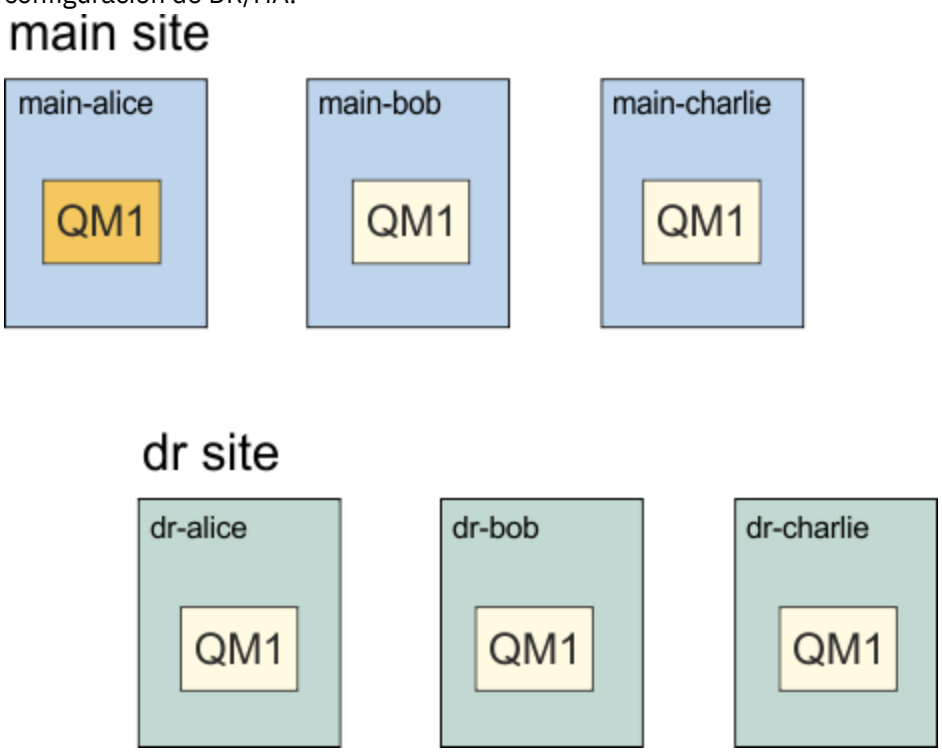
Tabla 36. Estado de nodo actual (continuación)

Atributo de estado	Valores posibles	Cuándo se muestra
Estado de DR	Normal Sincronización en curso Particionado Sistema remoto no disponible Incoherente Restaurando a la instantánea Sistema remoto no configurado Error en negociación	Todo es correcto. La sincronización está en curso. El usuario ha iniciado el gestor de colas en cada nodo cuando la red de réplica de DR no estaba disponible. La conexión con el otro nodo se ha perdido. Una sincronización estaba en curso pero se ha interrumpido. El usuario ha elegido volver a la instantánea que se tomó cuando el gestor de colas entró en estado incoherente. Se ha configurado el primario pero no el secundario. La negociación inicial entre los nodos primario y secundario ha fallado. Esto puede deberse a tipos de réplica incompatibles o a que el nodo secundario se haya configurado con un tamaño de sistema de archivos más pequeño.
Estado de DR (en nodo secundario de HA)	Consulte *Nodo_primario_HA*	Se muestra en los nodos secundarios de HA, ya que el estado de DR solo se conoce en el nodo primario de HA.
Puerto de DR	El puerto TCP/IP utilizado para replicar los datos para este gestor de colas.	Siempre se muestra.
Dirección IP local de DR	La dirección IP local que este gestor de colas utilizará para realizar la réplica de DR	Siempre se muestra.
Lista de direcciones IP remotas de DR	Las direcciones IP remotas que este gestor de colas utilizará para la réplica de DR. Una lista separada por comas de tres direcciones IP.	Siempre se muestra.
Dirección IP remota actual de DR	La IP remota actual a la que este gestor de colas está conectado para la réplica de DR.	Para un primario de HA con una conexión de DR activa.
Dirección IP remota actual de DR (en el nodo secundario de HA)	Consulte *Nodo_primario_HA*	Se muestra en un nodo secundario HA, ya que la conexión de DR solo está en el nodo primario de HA
Datos no sincronizados de DR	*x* KB	Se muestra cuando el nodo remoto no está disponible o no es coherente.
Progreso de sincronización de DR	*y* %	Se muestra cuando hay una sincronización en curso.

Tabla 36. Estado de nodo actual (continuación)		
Atributo de estado	**Valores posibles**	**Cuándo se muestra**
Hora estimada de finalización de DR	*aaaa-MM-dd HH:mm:ss*	Se muestra cuando hay una sincronización en curso.
Progreso de la reversión de instantánea	*y* %	Se muestra cuando el estado de DR es "Restaurando a la instantánea".
Última sincronización de DR	*aaaa-MM-dd HH:mm:ss*	Muestra cuando los datos de DR están sin sincronizar (después de la sincronización inicial). Proporciona la hora y la fecha cuando los datos se han sincronizado por última vez.

Ejemplo

Estos ejemplos ilustran la ejecución del mandato **`rdqmstatus -m qm1`** en varios nodos de la siguiente configuración de DR/HA:



Ejemplo de estado normal en un nodo que es el primario DR y el primario HA:

```
Node:                                 main-alice
Queue manager status:                 Running
CPU:                                  0.00%
Memory:                               123MB
Queue manager file system:            51MB used, 1.0GB allocated [5%]
HA role:                              Primary
HA status:                            Normal
HA control:                           Enabled
HA current location:                  This node
HA preferred location:                This node
HA blocked location:                  None
HA floating IP interface:             None
HA floating IP address:               None
DR role:                              Primary
DR status:                            Normal
DR port:                              3000
DR local IP address:                  192.168.1.1
DR remote IP address list:            192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address:         192.168.2.1
```

```
Node:                                   main-bob
HA status:                              Normal

Node:                                   main-charlie
HA status:                              Normal
```

Ejemplo de estado normal en un nodo que es el primario DR y un secundario HA:

```
Node:                                   main-bob
Queue manager status:                   Running elsewhere
HA role:                                Secondary
HA status:                              Normal
HA control:                             Enabled
HA current location:                    main-alice
HA preferred location:                  main-alice
HA blocked location:                    None
HA floating IP interface:               None
HA floating IP address:                 None
DR role:                                Primary
DR status:                              See main-alice
DR port:                                3000
DR local IP address:                    192.168.1.2
DR remote IP address list:              192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address:           See main-alice

Node:                                   main-alice
HA status:                              Normal

Node:                                   main-charlie
HA status:                              Normal
```

Ejemplo de estado normal en un nodo que es el secundario DR y un primario HA:

```
Node:                                   dr-alice
Queue manager status:                   Ended immediately
HA role:                                Primary
HA status:                              Normal
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA blocked location:                    None
HA floating IP interface:               None
HA floating IP address:                 None
DR role:                                Secondary
DR status:                              Normal
DR port:                                3000
DR local IP address:                    192.168.2.1
DR remote IP address list:              192.168.1.1,192.168.1.2,192.168.1.3
DR current remote IP address:           192.168.1.1

Node:                                   dr-bob
HA status:                              Normal

Node:                                   dr-charlie
HA status:                              Normal
```

Ejemplo de estado normal en un nodo que es el secundario DR y un secundario HA:

```
Node:                                   dr-bob
Queue manager status:                   Ended immediately
HA role:                                Secondary
HA status:                              Normal
HA control:                             Enabled
HA current location:                    dr-alice
HA preferred location:                  dr-alice
HA blocked location:                    None
HA floating IP interface:               None
HA floating IP address:                 None
DR role:                                Secondary
DR status:                              See dr-alice
DR port:                                3000
DR local IP address:                    192.168.2.2
DR remote IP address list:              192.168.1.1,192.168.1.2,192.168.1.3
DR current remote IP address:           See dr-alice

Node:                                   dr-alice
HA status:                              Normal
```

```
Node:                                   dr-charlie
HA status:                              Normal
```

Ejemplo de sincronización de DR en curso en un nodo que es un primario DR y un primario HA:

```
Node:                                   main-alice
Queue manager status:                   Running
CPU:                                    0.00%
Memory:                                 123MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
HA role:                                Primary
HA status:                              Normal
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA blocked location:                    None
HA floating IP interface:               None
HA floating IP address:                 None
DR role:                                Primary
DR status:                              Normal
DR port:                                3000
DR local IP address:                    192.168.1.1
DR remote IP address list:              192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address:           192.168.2.1
DR synchronization progress:            11.0%
DR estimated time to completion:        2018-09-06 14:55:05

Node:                                   main-bob
HA status:                              Normal

Node:                                   main-charlie
HA status:                              Normal
```

Ejemplo de DR particionada en un nodo que es un primario DR y un primario HA:

```
Node:                                   main-alice
Queue manager status:                   Running
CPU:                                    0.00%
Memory:                                 123MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
HA role:                                Primary
HA status:                              Normal
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA blocked location:                    None
HA floating IP interface:               None
HA floating IP address:                 None
DR role:                                Primary
DR status:                              Partitioned
DR port:                                3000
DR local IP address:                    192.168.1.1
DR remote IP address list:              192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address:           192.168.2.1
DR out of sync data:                    372KB

Node:                                   main-bob
HA status:                              Normal

Node:                                   main-charlie
HA status:                              Normal
```

Ejemplo de DR sin sincronizar de un nodo que es un primario DR y un primario HA:

```
Node:                                   main-alice
Queue manager status:                   Running
CPU:                                    0.00%
Memory:                                 123MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
HA role:                                Primary
HA status:                              Normal
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA blocked location:                    None
HA floating IP interface:               None
HA floating IP address:                 None
```

```
DR role:                                Primary
DR status:                              Remote unavailable
DR port:                                3000
DR local IP address:                    192.168.1.1
DR remote IP address list:              192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address:           Unknown
DR out of sync data:                    372KB
DR last in sync:                        2020-02-02 20:22:02

Node:                                   main-bob
HA status:                              Normal

Node:                                   main-charlie
HA status:                              Normal
```

Ejemplo de HA sin sincronizar de un nodo que es un primario DR y un primario HA:

```
Node:                                   main-alice
Queue manager status:                   Running
CPU:                                    0.00%
Memory:                                 123MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
HA role:                                Primary
HA status:                              Normal
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA blocked location:                    None
HA floating IP interface:               None
HA floating IP address:                 None
DR role:                                Primary
DR status:                              Normal
DR port:                                3000
DR local IP address:                    192.168.1.1
DR remote IP address list:              192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address:           192.168.2.1

Node:                                   main-bob
HA status:                              Inconsistent
HA out of sync data:                    15932KB
HA last in sync:                        2020-02-02 20:22:02

Node:                                   main-charlie
HA status:                              Normal
```

Ejemplo de un estado de resumen que muestra una discrepancia entre la versión del kernel del sistema operativo (RHEL 9.3) y el módulo de kernel DRBD (destinado a RHEL 9.2). Aunque el estado indica que se ha cargado el módulo de kernel de DRBD y se está ejecutando el gestor de colas, en esta situación debe actualizar el módulo de kernel de DRBD con la versión destinada al kernel del sistema operativo en ejecución.

```
Node:                                   main-alice
OS kernel version:                      5.14.0-362.18.1
DRBD OS kernel version:                 5.14.0-284.11.1
DRBD version:                           9.2.7+ptf.14
DRBD kernel module status:              Loaded

Queue manager name:                     QM1
Queue manager status:                   Running
HA current location:                    This node
HA preferred location:                  This node
HA blocked location:                    None
DR role:                                Primary
```

Ejemplo de un estado de resumen que muestra una discrepancia entre la versión de kernel del sistema operativo (RHEL 9.3) y el módulo de kernel DRBD (destinado a RHEL 9.0). En este ejemplo, la discrepancia de versiones es más grave y el módulo de kernel de DRBD no se puede cargar correctamente. QM1 es un gestor de colas HA/DR y se traslada a otro nodo, se desconoce su estado de HA y se desconoce su estado de DR. Para resolver esta anomalía, el módulo de kernel de DRBD debe actualizarse con la versión de destino para el kernel del sistema operativo en ejecución.

```
Node:                                   main-alice
OS kernel version:                      5.14.0-362.18.1
DRBD OS kernel version:                 5.14.0-70.13.1
DRBD version:                           9.2.7+ptf.14
```

```
DRBD kernel module status:              Partially loaded

Queue manager name:                     QM1
Queue manager status:                   Running elsewhere
HA status:                              Unknown
HA current location:                    main-bob
HA preferred location:                  This node
HA blocked location:                    None
DR role:                                Primary
DR status:                              Unknown
```

Referencia relacionada

Linux rdqmstatus

Linux *Funcionamiento en un entorno de DR/HA*

Al operar en un entorno de DR/HA hay consideraciones aparte para alta disponibilidad y recuperación tras desastre.

Si el nodo en el que se ejecuta un RDQM de DR/HA falla, el RDQM realiza automáticamente una migración tras error a otro nodo de ese grupo HA. Si falla todo el sitio, debe iniciar manualmente el RDQM en el nodo preferido del grupo HA en el sitio de recuperación. Las consideraciones aquí son las mismas que para un RDQM de DR corriente, consulte "Funcionamiento en un entorno de recuperación tras desastre" en la página 645 para obtener más información.

Si uno de los nodos falla por completo y debe sustituirse, consulte "Sustitución de un nodo que ha fallado en una configuración de recuperación tras desastre" en la página 647 y "Sustitución de un nodo que ha fallado en una configuración de disponibilidad" en la página 625 para obtener instrucciones.

Linux *Sustitución de un nodo anómalo en una configuración de DR/HA*

Si uno de los nodos de cualquiera de los grupos HA falla, puede sustituirlo.

Acerca de esta tarea

El procedimiento varía en función de si el nodo que va a sustituir es primario o secundario en la configuración de DR. En cualquier caso, el nodo nuevo debe tener una configuración idéntica al nodo que está sustituyendo, es decir, debe tener el mismo nombre de host, las mismas direcciones IP, etc.

También es posible que se encuentre en la situación en que ha perdido completamente el grupo de alta disponibilidad en el sitio principal o de recuperación y tiene que sustituir todo el grupo HA.

Procedimiento

- Para un nodo de sustitución que es primario en la configuración de DR, realice los pasos siguientes en el nodo nuevo:
  a) Cree un archivo `rdqm.ini` que coincida con los archivos de los demás nodos y a continuación ejecute el mandato `rdqmadm -c` (consulte "Definición del clúster de Pacemaker (grupo HA)" en la página 601).
  b) Ejecute el mandato `crtmqm -sxs -rr p` *qmanager* para volver a crear cada RDQM DR/HA (consulte "Creación de RDQM de DR/HA" en la página 653).
- Para un nodo de sustitución que es secundario en la configuración de DR, realice los pasos siguientes en el nodo nuevo:
  a) Cree un archivo `rdqm.ini` que coincida con los archivos de los demás nodos y a continuación ejecute el mandato `rdqmadm -c` (consulte "Definición del clúster de Pacemaker (grupo HA)" en la página 601).
  b) Ejecute el mandato `crtmqm -sx -rr s` *qmanager* para volver a crear cada RDQM DR/HA (consulte "Creación de RDQM de DR/HA" en la página 653).
- Para sustituir un grupo HA completo, realice los pasos siguientes:

a) Si pierde todo el grupo HA en el sitio primario de DR (es decir, el sitio principal), debe seguir los pasos para realizar una migración tras error gestionada al sitio secundario de DR para seguir ejecutando los RDQM DR/HA (consulte “Funcionamiento en un entorno de recuperación tras desastre” en la página 645). (Si pierde un grupo HA entero en el sitio de recuperación, los RDQM DR/HA continúan ejecutándose en el sitio principal sin su intervención).

b) Vuelva a crear el grupo HA en los tres nodos de sustitución, tal como se describe en “Configuración de grupos HA para RDQM DR/HA” en la página 652.

c) Vuelva a crear los RDQM DR/HA en el nuevo grupo HA como se describe en “Creación de RDQM de DR/HA” en la página 653.

d) Si es necesario, realice una migración tras error gestionada desde el sitio de recuperación de nuevo a su sitio principal.

Linux

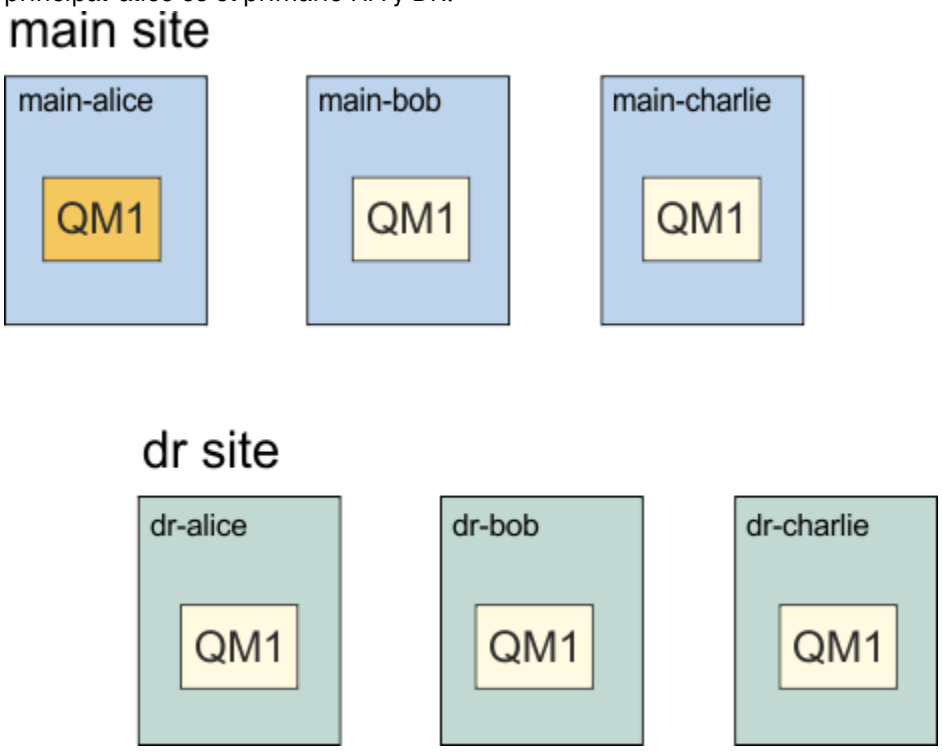
Ejemplo de trabajo de RDQM DR/HA

En este ejemplo se muestra cómo crear y suprimir un RDQM DR/HA.

Creación de un RDQM DR/HA

La configuración de ejemplo tiene dos sitios, denominados 'main' y 'dr'. Cada sitio tiene tres nodos, llamados 'alice', 'bob', y 'charlie'. Los nodos tienen un nombre completo que consta del nombre y el nombre de sitio, así 'main-alice', 'dr-alice', etc.

En los pasos siguientes se crea un RDQM DR/HA denominado QM1 que se ejecuta en main-alice. El nodo principal-alice es el primario HA y DR.



Si las direcciones IP de DR local y remota se especifican en el archivo `rdqm.ini`, no es necesario especificar ninguna dirección IP en la línea de mandatos y se puede crear un RD/HA RDQM denominado QM1 ejecutando el mandato siguiente en el alice principal:

```
crtmqm -sx -rr p -rn DR1 -rp 7001 QM1
```

Si las direcciones IP de DR local se especifican en el archivo `rdqm.ini`, las direcciones IP de DR remoto se pueden especificar en la línea de mandatos:

```
crtmqm -sx -rr p -ri 192.168.2.1,192.168.2.2,192.168.2.3 -rp 7001 QM1
```

Si no se especifican direcciones IP de DR en el archivo `rdqm.ini`, se pueden especificar las direcciones IP remotas y locales DR en la línea de mandatos:

```
crtmqm -sx -rr p -rl 192.168.1.1,192.168.1.2,192.168.1.3 -ri
192.168.2.1,192.168.2.2,192.168.2.3 -rp 7001 QM1
```

En el ejemplo siguiente, se muestra la salida en respuesta a la creación de QM1:

```
Creating replicated data queue manager configuration.
Secondary queue manager created on 'main-bob'.
Secondary queue manager created on 'main-charlie'.
IBM MQ queue manager created.
Directory '/var/mqm/vols/qm1/qmgr/qm1' created.
The queue manager is associated with installation 'Installation1'.
Creating or replacing default objects for queue manager 'QM1'.
Default objects statistics : 83 created. 0 replaced. 0 failed.
Completing setup.
Setup completed.
Enabling replicated data queue manager.
Replicated data queue manager enabled.
Issue the following command on the remote HA group to create the DR/HA secondary queue manager:
crtmqm -sx -rr s -rl 192.168.2.1,192.168.2.2,192.168.2.3 -ri
192.168.1.1,192.168.1.2,192.168.1.3 -rp 7001 -fs 3072M QM1
```

Copie el mandato del mensaje para crear la instancia secundaria DR de QM1 en dr-alice:

```
crtmqm -sx -rr s -rl 192.168.2.1,192.168.2.2,192.168.2.3 -ri
192.168.1.1,192.168.1.2,192.168.1.3 -rp 7001 -fs 3072M QM1
```

El mensaje siguiente es la salida en dr-alice:

```
Creating replicated data queue manager configuration.
Secondary queue manager created on 'dr-bob'.
Secondary queue manager created on 'dr-charlie'.
IBM MQ secondary queue manager created.
Enabling replicated data queue manager.
```

Prueba de la DR secundaria

Para probar las características de recuperación tras desastre de QM1, ejecute el mandato siguiente en main-alice para hacer que QM1 sea la instancia secundaria de DR:

```
rdqmdr -m QM1 -s
Queue manager 'QM1' has been made the DR secondary on this node.
```

Ejecute el mandato siguiente en dr-alice para hacer que QM1 sea la instancia primaria de DR en ese nodo:

```
rdqmdr -m QM1 -p
Queue manager 'QM1' has been made the DR primary on this node.
```

Supresión de un RDQM DR/HA

Para suprimir el RDQM DR/HA denominado QM1, primero finalice el gestor de colas en main-alice:

```
endmqm -w QM1
Replicated data queue manager disabled.
Waiting for queue manager 'QM1' to end.
IBM MQ queue manager 'QM1' ended.
```

A continuación, ejecute el mandato siguiente en main-alice para suprimir QM1:

```
dltmqm QM1
Removing replicated data queue manager configuration.
Secondary queue manager deleted on 'main-bob'.
Secondary queue manager deleted on 'main-charlie'.
IBM MQ queue manager 'QM1' deleted.
```

Finalmente, debe suprimir QM1 en dr-alice:

```
dltmqm QM1
Removing replicated data queue manager configuration.
Secondary queue manager deleted on 'dr-bob'.
Secondary queue manager deleted on 'dr-charlie'.
IBM MQ queue manager 'QM1' deleted.
```

Conceptos relacionados

“Funcionamiento en un entorno de DR/HA” en la página 668
Al operar en un entorno de DR/HA hay consideraciones aparte para alta disponibilidad y recuperación tras desastre.

Tareas relacionadas

“Creación de RDQM de DR/HA” en la página 653
Puede utilizar el mandato **crtmqm** para crear un gestor de colas de datos replicados (RDQM) en una configuración DR/HA.

“Supresión de un RDQM DR/HA” en la página 657
Puede utilizar el mandato **dltmqm** para suprimir un gestor de colas de datos replicados (RDQM) de DR/HA.

CP4I MQ Adv. HA nativa

La alta disponibilidad nativa es una solución de alta disponibilidad que está disponible en despliegues de contenedor de IBM MQ.

Una configuración de HA nativa consta de tres nodos (que pueden ser, por ejemplo, tres pods Kubernetes), cada uno con una instancia del gestor de colas. Una instancia es el gestor de colas activo, procesando mensajes y grabando en su registro. Siempre que se graba el registro, el gestor de colas activo envía los datos a las otras dos instancias, conocidas como 'réplicas'. Cada réplica graba en su propio registro, reconoce los datos y, a continuación, actualiza sus propios datos de cola del registro replicado. Si el nodo que ejecuta el gestor de colas activo falla, una de las instancias de réplica del gestor de colas toma el control del rol activo y tiene datos actuales con los que operar.

Para obtener una visión general detallada, consulte HA nativa en la sección Contenedores de esta documentación.

La figura siguiente muestra un despliegue típico con tres instancias de un gestor de colas desplegadas en tres contenedores.

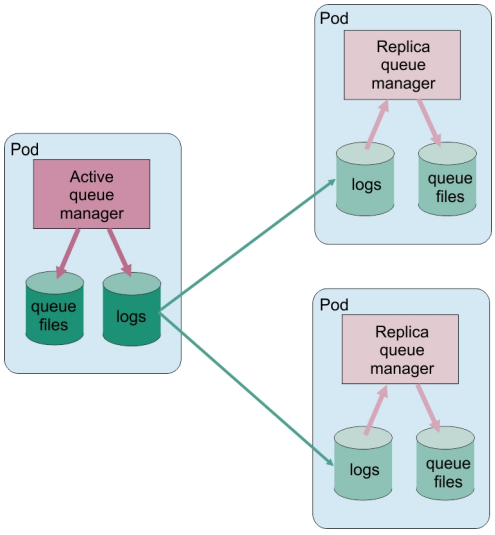


Figura 83. Ejemplo de configuración de HA nativa

CP4I MQ Adv. *Creación de la solución de HA nativa*

El método recomendado para crear una solución de HA nativa es utilizar IBM MQ Operator. De forma alternativa, puede crear sus propios contenedores y configurar manualmente la HA nativa.

Nota: Esta información sólo se aplica a entornos de contenedor.

Para crear una solución de HA nativa utilizando la IBM MQ Operator, consulte HA nativa para obtener una visión general y Ejemplo: Configuración de un gestor de colas de HA nativa para obtener instrucciones detalladas.

Para crear sus propios contenedores y configurar manualmente la HA nativa, consulte Creación del grupo HA nativa si crea sus propios contenedores.

CP4I MQ Adv. *Finalización de gestores de colas de HA nativa*

Para IBM MQ en contenedores, puede utilizar el mandato **endmqm** para finalizar un gestor de colas activo o de réplica que forme parte de un grupo HA nativo.

Acerca de esta tarea

Nota: Esta información sólo se aplica a entornos de contenedor.

El procedimiento para detener un gestor de colas que forma parte de un grupo HA nativo depende de si es una instancia activa o de réplica. Cuando finaliza cualquiera de los dos tipos de instancia, se realiza una comprobación para asegurarse de que la finalización de la instancia no interrumpe el quórum del grupo HA nativo. Si se rompe el quórum, el mandato **endmqm** falla.

Cuando emite un mandato **endmqm** , se avisa a las otras instancias del grupo de que esto está ocurriendo, para que no notifique errores cuando se interrumpa la conexión.

Si una instancia activa pierde quórum debido a la finalización o desconexión de demasiadas instancias de réplica, la instancia activa espera durante un periodo de tiempo configurable antes de finalizar completamente. Esto permite un periodo de tiempo para concluir el proceso correctamente, en lugar de que las aplicaciones simplemente tengan sus conexiones interrumpidas. Este valor de tiempo de espera se puede especificar mediante el atributo `QuorumConnectivityTimeout` en la stanza `NativeHALocalInstance` del archivo `qm.ini` . El valor predeterminado es 0 segundos.

Procedimiento

- Para finalizar la instancia activa de un gestor de colas, emita el mandato siguiente en el nodo donde se ejecuta la instancia activa:

  ```
  endmqm -s QMgrName
  ```

  – Especifique la opción `-r` para ayudar a las aplicaciones cliente a reconectarse a otra instancia.
  – Si esta instancia no es la instancia activa en el grupo HA nativo, el mandato falla.
  – Si la finalización de esta instancia activa hace que falle el quórum de grupo, el mandato falla. (Si otras instancias finalizan o dejan de estar disponibles al mismo tiempo que ejecuta este mandato, es posible que la comprobación de quórum no lo detecte, el grupo HA nativo finaliza y solo se puede reiniciar cuando haya suficientes instancias disponibles.)

  Cuando el gestor de colas activo finaliza, una de las instancias de réplica toma el control del rol activo. No puede especificar qué réplica toma el control, esto lo determina la negociación dentro del grupo y depende de cuál tenga los registros de transacciones más actualizados.
- Para finalizar una instancia de réplica de un gestor de colas, emita el mandato siguiente:

  ```
  endmqm -x QMgrName
  ```

  – Si esta instancia es la instancia activa, el mandato falla.
  – Si la finalización de esta instancia de réplica hace que falle el quórum de grupo, el mandato falla. (Si otras instancias finalizan o dejan de estar disponibles al mismo tiempo que ejecuta este mandato,

es posible que la comprobación de quórum no lo detecte, el grupo HA nativo finaliza y solo se puede reiniciar cuando haya suficientes instancias disponibles.)

Nota: También puede utilizar los conmutadores -c, -i, -p o -w con el mandato **endmqm** en instancias de HA nativa, independientemente del rol en el que se encuentren. La instancia del gestor de colas finaliza, ignorando el efecto que tiene en el quórum del grupo. Sin embargo, la información se sigue compartiendo con las otras instancias del grupo. Puede utilizar estos conmutadores junto con -s para la instancia activa. No puede utilizar estos conmutadores junto con el conmutador -x para instancias de réplica.

Referencia relacionada

endmqm (finalizar gestor de colas)

Registro: Asegurarse de que no se han perdido mensajes

IBM MQ registra todos los cambios significativos en los datos persistentes controlados por el gestor de colas en un registro de recuperación.

Esto incluye la creación y supresión de objetos, las actualizaciones de mensajes persistentes, los estados de transacciones, los cambios realizados en atributos de objetos y las actividades de canal. El registro contiene la información que se necesita para recuperar todas las actualizaciones en colas de mensajes. Para ello:

- Mantiene registros de los cambios del gestor de colas.
- Mantiene registros de las actualizaciones de las colas para que los utilice el proceso de reinicio.
- Le permite restaurar datos después de una anomalía de hardware o de software.

Sin embargo, IBM MQ también depende del sistema de discos que aloja los archivos, incluidos los archivos de registro. Si el sistema de discos no es fiable en sí mismo, se puede perder información, incluida la información de registro.

PRECAUCIÓN: No puede mover los registros de recuperación a un sistema operativo distinto.

Cómo son los registros

Los registros constan de archivos primarios y secundarios y de un archivo de control. El usuario define el número y tamaño de los archivos de registro y el lugar donde se almacenan en el sistema de archivos.

Un registro de IBM MQ consta de dos componentes:

1. Uno o varios archivos de datos de anotaciones.
2. Un archivo de control de anotaciones.

Un archivo de datos de anotaciones también se conoce como extensión de anotaciones.

Hay varias extensiones de registro que contienen los datos que se están registrando. Puede definir el número y el tamaño (como se explica en “Stanza LogDefaults del archivo mqs.ini” en la página 105) o tomar el valor predeterminado de sistema de tres extensiones primarias y de dos secundarias.

Cada una de los tres extensiones primarias y las dos secundarias toman de forma predeterminada 16 MB.

Cuando se crea un gestor de colas, el número de extensiones de registro preasignadas es el número de extensiones de registro *primarias* asignadas. Si no se especifica ningún número, se utilizará el valor predeterminado.

IBM MQ utiliza dos tipos de registro:

- Circular
- Lineal

  El número de extensiones de registro utilizadas con el registro lineal puede ser muy grande, dependiendo de la frecuencia de la grabación de imagen de soporte

Consulte “Tipos de registro” en la página 674 para obtener más información.

ALW En los sistemas IBM MQ for AIX or Linux, si no ha cambiado la vía de acceso de registro, las extensiones de registro se crean en el directorio:

```
/var/mqm/log/QMgrName
```

Windows En IBM MQ for Windows, si no ha cambiado la vía de acceso de registro, se crean extensiones de registro bajo el directorio:

```
C:\ProgramData \IBM \MQ\log\QMgrName
```

IBM MQ se inicia con estas extensiones de registro primario, pero si el espacio de registro primario no es suficiente, asigna extensiones de registro *secundario*. Lo hace de forma dinámica y los elimina cuando disminuye la necesidad de espacio de registro. De forma predeterminada, se pueden asignar hasta dos extensiones de registro secundario. Puede cambiar esta asignación predeterminada, tal como se describe en “Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms” en la página 95.

A las extensiones de registro se les añade como prefijo la letra S o la letra R. A las extensiones activas, inactivas y superfluas se les añade el prefijo S, mientras que a las extensiones de reutilización se les añade el prefijo R.

Cuando se realiza una copia de seguridad o se restaura el gestor de colas, se hace una copia de seguridad y se restauran todas las extensiones activas, inactivas y superfluas, junto con el archivo de control de registro.

Nota: No es necesario hacer copias de seguridad y restaurar las extensiones de reutilización.

El archivo de control de registros

El archivo de control de registros contiene la información necesaria para describir el estado de extensiones de registro, como por ejemplo su tamaño y ubicación y el nombre de la siguiente extensión disponible.

Importante: El archivo de control de registro es solo para uso interno del gestor de colas.

El gestor de colas mantiene los datos de control asociados con el estado del registro de recuperación en el archivo de control de registro y no debe modificar el contenido del archivo de control de registro.

El archivo de control de registro está en la vía de acceso de registro y se denomina `amqhlctl.lfh`. Cuando realice una copia de seguridad o restaure el gestor de colas, asegúrese de que se haga una copia de seguridad y se restaure el archivo de control de registro, junto con las extensiones de registro.

Tipos de registro

En IBM MQ, hay dos formas de mantener los registros de las actividades del gestor de colas: el registro circular y el registro lineal. Un tercer tipo de registro, replicado, sólo lo utilizan las configuraciones de HA nativa.

Anotaciones cronológicas circulares

Utilice el registro circular si todo lo que desea hacer es la recuperación de reinicio, utilizando el registro para restituir las transacciones que estaban en curso cuando se detuvo el sistema.

El registro de anotaciones circular mantiene todos los datos de reinicio en un anillo de archivos de anotaciones. El registro cronológico llena el primer archivo del anillo, luego pasa al siguiente y así sucesivamente hasta que se llenan todos los archivos. Después, vuelve al primer archivo del anillo y empieza de nuevo. Este proceso continúa mientras el producto está utilizándose y tiene la ventaja de que el usuario nunca se queda sin archivos de registro.

IBM MQ guarda las entradas de registro necesarias para reiniciar el gestor de colas sin pérdida de datos hasta que dejan de ser necesarias para asegurar la recuperación de los datos del gestor de colas. El mecanismo para liberar los archivos de registro para su reutilización se describe en "Utilización de la sincronización por puntos de comprobación para asegurar la recuperación completa" en la página 676.

Anotaciones cronológicas lineales

Utilice el registro lineal si desea realizar tanto la recuperación de reinicio como la recuperación desde soporte (volver a crear los datos dañados o perdidos reproduciendo el contenido del registro). El registro lineal mantiene los datos de registro en una secuencia continua de archivos de registro.

Los archivos de registro también pueden:

- Reutilizarse, pero solo cuando ya no son necesarios para la recuperación de reinicio o la recuperación de soportes.
- Archivarse manualmente para el almacenamiento y el análisis a largo plazo.

La frecuencia de las imágenes de soporte determina cuándo pueden reutilizarse los archivos de registro lineal y es un factor importante para determinar cuánto espacio de disco debe estar disponible para los archivos de registro lineal.

Puede configurar el gestor de colas para crear automáticamente imágenes de soporte periódicas, basándose en el tiempo o el uso de registro, o puede planificar las imágenes de soporte manualmente.

El administrador decide qué política se implementa y las implicaciones en el uso del espacio de disco. Los archivos de registro necesarios para la recuperación de reinicio siempre deben estar disponibles, mientras que los archivos de registro necesarios solo para la recuperación de soportes pueden archivarse en un almacenamiento a largo plazo, por ejemplo, en cinta.

Si el administrador permite la gestión de registro automática y las imágenes de soporte automáticas, el registro lineal se comporta de forma parecida a un registro circular muy grande, pero con la redundancia mejorada contra errores de soporte habilitada por la recuperación de soportes.

Puede cambiar un tipo de registro existente para un gestor de colas, de lineal a circular, o de circular a lineal utilizando el mandato migmqlog .

Registro replicado

CP4I

Utilice el registro replicado para configurar una configuración de HA nativa. Al crear un grupo de HA nativa, crea tres gestores de colas en distintos nodos. Especifica un tipo de registro de replicado junto con un nombre de instancia exclusivo para cada uno de los gestores de colas. La configuración de HA nativa proporciona una solución de alta disponibilidad ya que tiene una instancia activa que replica los datos de registro en dos instancias de réplica. Si la instancia activa falla, una de las instancias de réplica asume el rol activo. La réplica de registro garantiza que no se pierden apenas datos, si se pierden algunos. Consulte "HA nativa" en la página 671 para obtener más detalles. Un registro replicado es equivalente a un registro lineal con la gestión automática de registros y las imágenes de soporte automáticas habilitadas.

Extensiones de registro lineal que no están activas

Multi

Si utiliza la gestión automática de registros, incluido el archivado, el registrador realiza un seguimiento de las extensiones de registro lineales que no están activas.

Atención: Si utiliza la gestión de registro automática sin el archivado, el uso de un gestor de colas de copia de seguridad no está soportado para este proceso.

ALW Cuando una extensión de registro ya no es necesaria para la recuperación y, si es necesario, se archiva, el registrador suprimirá la extensión de registro o la reutilizará en un determinado momento, cuando sea conveniente.

A una extensión de registro reutilizada se le cambia el nombre para que sea la siguiente en la secuencia de registro. El mensaje AMQ7490 se graba periódicamente, indicando cuántas extensiones se han creado, suprimido o reutilizado.

El registrador decide cuántas extensiones debe mantener listas para su reutilización y cuándo se deben suprimir esas extensiones.

Registro activo

Hay varios archivos que se consideran *activos* en el registro lineal y circular. El registro activo es la cantidad máxima de espacio de registro, si está utilizando el registro circular o lineal, que puede ser referenciado por la recuperación de reinicio.

El número de archivos de registro activos suele ser inferior al número de archivos de registro primarios definido en los archivos de configuración. (Consulte "Cálculo del tamaño del registro" en la página 680 para obtener información sobre cómo definir el número).

Tenga en cuenta que el espacio de registro activo no incluye el espacio necesario para la recuperación de soporte y que el número de archivos de registro utilizado con el registro lineal puede ser muy grande, dependiendo del flujo de mensajes y la frecuencia de las imágenes de soporte.

Registro inactivo

Cuando un archivo de registro ya no es necesario para la recuperación de reinicio, cambia su estado a *inactive*. Los archivos de registro que no son necesarios para la recuperación de reinicio o la recuperación de soportes pueden considerarse archivos de registro superfluos.

Cuando se utiliza la gestión de registro automática, el gestor de colas controla el proceso de estos archivos de registro superfluos. Si ha seleccionado la gestión de registro manual, será responsabilidad del administrador gestionar (por ejemplo, suprimir y archivar) los archivos de registro superfluos si ya no son relevantes para la operación.

Consulte "Gestión de registros" en la página 686 si desea obtener más información sobre la eliminación de archivos de registro.

Archivos de registro secundarios

Aunque los archivos de registro secundarios se definen para el registro lineal, no se utilizan en el funcionamiento normal. Si se presenta el caso en el que, probablemente debido a transacciones que están activas durante mucho tiempo, no sea posible liberar un archivo de la agrupación activa porque todavía podría ser necesario para un reinicio, se formatean y añaden archivos secundarios a la agrupación de archivos de registro activos.

Si se ha agotado el número de archivos secundarios disponibles, las solicitudes de operaciones adicionales que requieran actividad de registro se rechazarán, se devolverá el código de retorno MQRC_RESOURCE_PROBLEM a la aplicación y las transacciones de larga ejecución se considerarán para la retrotracción asíncrona.

Atención: Todos los tipos de registro pueden hacer frente a una pérdida de alimentación imprevista, suponiendo que no haya ninguna anomalía de hardware.

Utilización de la sincronización por puntos de comprobación para asegurar la recuperación completa

Los gestores de colas de registro circular y registro lineal dan soporte a la recuperación de reinicio. Aunque la instancia anterior del gestor de colas termine abruptamente (por ejemplo, por un apagón), cuando se reinicia, el gestor de colas restaura su estado persistente al estado transaccional correcto en el punto de terminación.

La recuperación de reinicio depende de que se mantenga la integridad del disco. De forma parecida, el sistema operativo debe garantizar la integridad de disco, aunque el sistema operativo termine abruptamente.

En el caso excepcional de que la integridad del disco no se mantenga, el registro lineal (y la recuperación de soportes) proporciona algunas opciones de redundancia y recuperación adicionales. Con una tecnología cada vez más común como RAID, es cada vez más raro sufrir problemas de integridad de disco y muchas empresas configuran el registro circular y solo utilizan la recuperación de reinicio.

IBM MQ está diseñado como un gestor de recursos de registro de escritura anticipada clásico. Las actualizaciones persistentes realizadas en las colas de mensajes se producen en dos etapas:

1. Los registros que representan la actualización se graban de forma fiable en el registro de recuperación.
2. El archivo de cola o los almacenamientos intermedios se actualizan de forma más eficaz para el sistema, pero no necesariamente de forma coherente.

Por lo tanto, los archivos de registro pueden estar más actualizados que el almacenamiento intermedio de cola y el estado de archivo subyacente.

Si esta situación se ha permitido que continúe sin cambios, se necesitará un volumen muy grande de ejecución de registros para que el estado de la cola sea coherente después de una recuperación tras bloqueo.

IBM MQ utiliza `checkpoints` para limitar el volumen de reproducción de registro necesario después de una recuperación tras bloqueo. El suceso clave que controla si un archivo de registro se denomina activo o no es un `checkpoint`.

Un punto de comprobación de IBM MQ es un punto:

- De coherencia entre el registro de recuperación y los archivos de objeto.
- Que identifica un lugar en el registro desde el que se garantiza la ejecución de los registros posteriores para restaurar la cola al estado lógico correcto en el momento en que el gestor de colas ha finalizado.

Durante un punto de comprobación, IBM MQ vacía las actualizaciones más antiguas en los archivos de cola, según sea necesario, para limitar el volumen de los registros que deben ejecutarse para devolver las colas a un estado coherente después de una recuperación tras bloqueo.

El punto de comprobación completo más reciente marca el punto en el registro desde el que debe ejecutarse la reproducción durante una recuperación de anomalía. Por lo tanto, la frecuencia del punto de comprobación es un equilibrio entre la sobrecarga del registro de puntos de comprobación y la mejora en el tiempo de recuperación potencial que ofrecen esos puntos de comprobación.

El registrador planifica los puntos de comprobación con más frecuencia (por lo que el siguiente se planifica antes de que se haya completado el anterior) porque el registrador está intentando mantener el registro activo en las extensiones de registro primario. Si esto no es posible, se registra un error AMQ7466.

La posición en el registro del inicio del punto de comprobación completo más reciente es uno de los factores clave a la hora de determinar si un archivo de registro está activo o inactivo. El otro factor clave es la posición en el registro del primer registro respecto a la primera actualización persistente realizada por una transacción activa actual.

Si se registra un nuevo punto de comprobación en el segundo archivo de registro, o en uno posterior, y ninguna transacción actual hace referencia a un registro en el primer archivo de registro, el primer archivo de registro se inactiva. En el caso de un registro circular, el primer archivo de registro está ahora listo para reutilizarse. En el caso de un registro lineal, el primer archivo de registro normalmente continuará siendo necesario para la recuperación de soportes.

Si configura el registro circular o la gestión de registros automática, el gestor de colas gestionará los archivos de registro inactivos. Si configura el registro lineal con la gestión de registros manual, es una tarea de administración gestionar los archivos inactivos según los requisitos de su operación.

IBM MQ genera puntos de comprobación automáticamente. Estos se toman en las ocasiones siguientes:

- Cuando se inicia el gestor de colas
- En el cierre
- Cuando queda poco espacio de registro

- Multi Después de que se registren 50.000 operaciones desde que se haya generado el punto de comprobación anterior
- z/OS Cuando se han registrado *número de operaciones* desde que se tomó el punto de comprobación anterior, donde *número de operaciones* es el número de operaciones establecido en la propiedad **LOGLOAD**.

Cuando IBM MQ se reinicia, busca 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. Se ejecutan todas las operaciones que han tenido lugar desde el punto de comprobación. Esto se conoce como fase de ejecución.

La fase de ejecución devuelve las colas al estado lógico en el que estaban antes de la anomalía o la conclusión del sistema. Durante la fase de ejecución, se crea una lista de las transacciones que estaban en curso cuando se produjo la anomalía o la conclusión del sistema.

Multi Se emiten los mensajes AMQ7229 and AMQ7230 para indicar la progresión de la fase de reproducción.

Para saber qué operaciones debe restituir o confirmar, IBM MQ accede a todos los registros de anotaciones activos asociados a una transacción en curso. Esto se conoce como la fase de recuperación.

Multi Se imiten los mensajes AMQ7231, AMQ7232 y AMQ7234 para indicar la progresión de la fase de recuperación.

Cuando se ha accedido a todos los registros de anotaciones necesarios durante la fase de recuperación, se resuelven las transacciones activas y las operaciones asociadas a la transacción se restituyen o se confirman. Esto se conoce como fase de resolución.

Multi Se emite el mensaje AMQ7233 para indicar la progresión de la fase de resolución.

z/OS En z/OS, el proceso de reinicio se compone de varias fases.

1. El rango de registro de recuperación se establece basándose en la recuperación de soportes necesaria para los conjuntos de páginas y el registro más antiguo que se necesita para restituir las unidades de trabajo y obtener los bloqueos para las unidades de trabajo pendientes.
2. Una vez determinado el rango de registro, se realiza la lectura de registro en adelante para llevar los conjuntos de páginas al último estado, así como para bloquear los mensajes relacionados con unidades de trabajo dudosas o en curso.
3. Cuando finaliza la lectura de registros en adelante, los registros se leen hacia atrás para restituir las unidades de trabajo que estaban en curso o en retroceso en el momento de la anomalía.

z/OS Un ejemplo de los mensajes que puede ver:

```
CSQR001I +MQOX RESTART INITIATED
CSQR003I +MQOX RESTART - PRIOR CHECKPOINT RBA=00000001E48C0A5E
CSQR004I +MQOX RESTART - UR COUNTS -  806
IN COMMIT=0, INDOUBT=0, INFLIGHT=0, IN BACKOUT=0
CSQR030I +MQOX Forward recovery log range  815
from RBA=00000001E45FF7AD to RBA=00000001E48C1882
CSQR005I +MQOX RESTART - FORWARD RECOVERY COMPLETE -  816
IN COMMIT=0, INDOUBT=0
CSQR032I +MQOX Backward recovery log range  817
from RBA=00000001E48C1882 to RBA=00000001E48C1882
CSQR006I +MQOX RESTART - BACKWARD RECOVERY COMPLETE -  818
INFLIGHT=0, IN BACKOUT=0
CSQR002I +MQOX RESTART COMPLETED
```

Nota: Si hay una gran cantidad de registros pendientes de lectura, se emiten los mensajes CSQR031I (recuperación hacia adelante) y CSQR033I (recuperación hacia atrás) periódicamente para mostrar el progreso.

En la Figura 84 en la página 679, IBM MQ ya no necesita ninguno de los registros anteriores al último punto de comprobación, Punto de comprobación 2. Las colas pueden recuperarse a partir de la información del punto de comprobación y de todas las entradas de registro posteriores. En el registro circular, todos los archivos liberados antes del punto de comprobación se pueden reutilizar. En un registro lineal, ya no es necesario acceder a los archivos de registro liberados para la operación normal y pasan a estar inactivos. En el ejemplo, el puntero de cabecera de cola se mueve para apuntar al último punto de comprobación, Checkpoint 2, que se convierte en la nuevo cabecera de cola, Head 2. El archivo de registro 1 ahora se puede volver a utilizar.

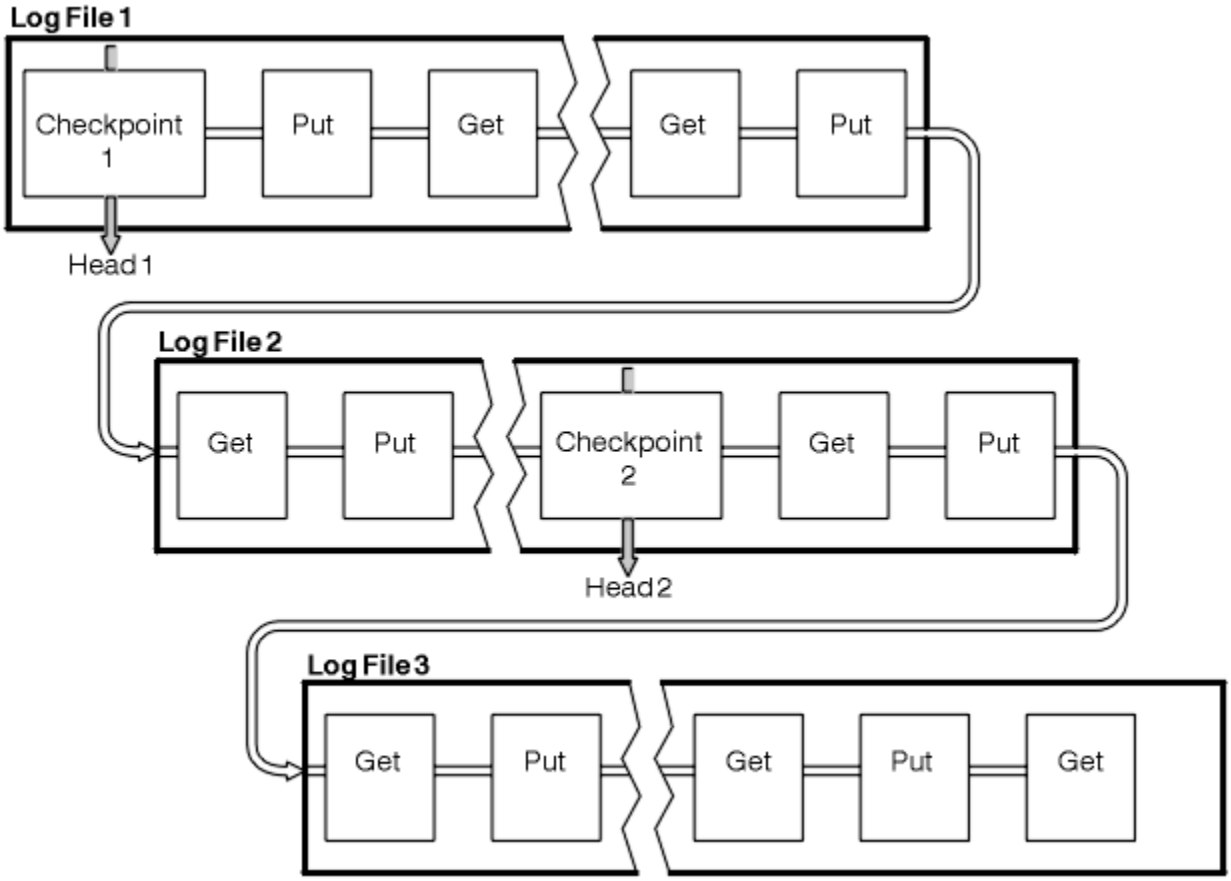


Figura 84. Sincronización por puntos de comprobación

Sincronización por puntos de comprobación con transacciones de larga ejecución

La forma en que una transacción de larga duración afecta a la reutilización de los archivos de anotaciones.

La Figura 85 en la página 680 muestra la forma en que una transacción de larga duración afecta a la reutilización de los archivos de registro. En el ejemplo, una transacción de larga ejecución ha realizado una entrada en el registro, representada como LR 1, después del primer punto de comprobación mostrado. La transacción no finaliza, (en el punto LR 2), hasta después del tercer punto de comprobación. Toda la información de registro, desde LR 1 en adelante, se retiene para permitir la recuperación de dicha transacción, si es necesario, hasta que finaliza.

Cuando la transacción de larga ejecución ha finalizado, en LR 2, la cabecera del registro se mueve lógicamente al punto de comprobación 3, el último punto de comprobación registrado. Los archivos que contienen registros de anotaciones anteriores al punto de comprobación 3, Cabecera 2, ya no son necesarios. Si está utilizando el registro circular, el espacio puede reutilizarse.

Si los archivos de registro primarios se llenan por completo antes de que finalice la transacción de larga ejecución, se pueden utilizar archivos de registro secundario para evitar que se llenen los registros.

Las actividades que están totalmente bajo el control del gestor de colas, por ejemplo, los puntos de comprobación, están planificadas para probarse y mantener la actividad en el registro primario.

No obstante, cuando el espacio de registro secundario debe dar soporte al comportamiento fuera del control del gestor de colas (por ejemplo, la duración de una de las transacciones), el gestor de colas intenta utilizar cualquier espacio de registro secundario definido, para que la actividad pueda completarse.

Si la actividad no se ha completado para cuando se está utilizando el 80% del espacio de registro total, el gestor de colas inicia una acción para reclamar el espacio de registro, independientemente del impacto que tenga en la aplicación.

Cuando la cabecera del registro se traslada y se está utilizando el registro circular, los archivos de registro primarios pueden seleccionarse para su reutilización y el registrador, después de llenar el archivo actual, reutiliza el primer archivo primario disponible. Si está utilizando el registro lineal, la cabecera del registro se mueve hacia el final de la agrupación activa y el primer archivo pasa a estar inactivo. Se formatea un nuevo archivo primario y se añade al final de la agrupación donde queda preparado para futuras actividades de registro cronológico.

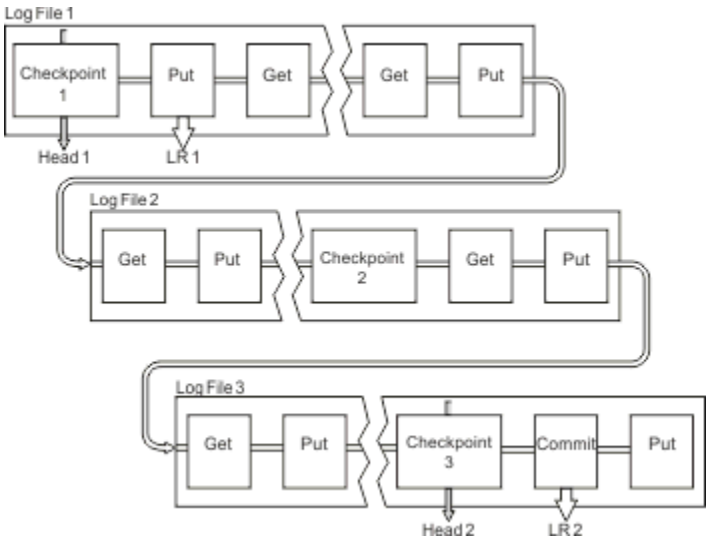


Figura 85. Sincronización por puntos de control en una transacción de larga ejecución

Cálculo del tamaño del registro

Cálculo del tamaño de las anotaciones cronológicas que un gestor de colas necesita.

Después de decidir si el gestor de colas utiliza el registro circular o lineal, necesita estimar el tamaño del Registro activo que gestor de colas necesita. El tamaño del registro activo lo determinan los siguientes parámetros de configuración de registro:

LogFilePages
El tamaño de cada archivo de registro primario y secundario en unidades de páginas de 4K.

LogPrimaryFiles
El número de archivos de anotaciones primarios preasignados

LogSecondaryFiles
El número de archivos de registro secundarios que pueden crearse para utilizarse cuando los archivos de registro primarios se están llenando

Notas:

1. Puede cambiar el número de archivos de registro primarios y secundarios cada vez que se inicia el gestor de colas, aunque puede que no observe el efecto de los cambios que realice en los registros secundarios inmediatamente.
2. No se puede cambiar el tamaño del archivo de registro; debe determinarlo **antes** de crear el gestor de colas.
3. El número de archivos de registro primarios y el tamaño del archivo de registro determinan la cantidad de espacio de registro preasignada al crear el gestor de colas.

4. El número total de archivos de registro primarios y secundarios no puede exceder de 511 en los sistemas AIX and Linux, o de 255 en los sistemas Windows, lo que en presencia de transacciones de larga ejecución, limita la cantidad máxima de espacio de registro disponible en el gestor de colas para la recuperación de reinicio. La cantidad de espacio de registro que el gestor de colas puede necesitar para la recuperación desde soporte no tiene este mismo límite.
5. Cuando se utiliza el registro *circular*, el gestor de colas reutiliza el espacio de registro primario y secundario. El gestor de colas asignará, hasta un determinado límite, un archivo de registro secundario cuando un archivo de registro se llene y el siguiente archivo de registro primario de la secuencia no esté disponible.

   Consulte "¿Qué tamaño debe tener el registro activo?" en la página 681 para obtener información sobre el número de registros que necesita asignar. Las extensiones de registro primario se utilizan en secuencia y esa secuencia no cambia.

   Por ejemplo, si tiene tres registros primarios 0, 1 y 2, el orden de uso es 0,1,2 seguido de 1,2,0, 2,0,1, volviendo a 0,1,2 y así sucesivamente. Los registros secundarios asignados se intercalan como es necesario.
6. Los archivos de registro primarios se pueden reutilizar durante un punto de comprobación. El gestor de colas tiene en cuenta el espacio de anotaciones primario y secundario antes de un punto de control debido a que queda poco espacio para anotaciones.

   El gestor de colas intenta planificar puntos de comprobación de una forma que mantiene el uso del registro dentro de las extensiones primarias.

Consulte "Stanza LogDefaults del archivo mqs.ini" en la página 105 para obtener más información.

¿Qué tamaño debe tener el registro activo?

Estimación del tamaño del registro activo que necesita un gestor de colas.

El tamaño del registro activo es limitado por lo siguiente:

```
logsize = (primaryfiles + secondaryfiles) * logfilepages * 4096
```

El registro debe ser lo suficientemente grande para hacer frente a la transacción de más larga ejecución que se ejecute cuando el gestor de colas esté grabando la cantidad máxima de datos por segundo en el disco.

Si la transacción de más larga ejecución se ejecuta durante N segundos y la cantidad máxima de datos por segundo grabados en disco por el gestor de colas es de B bytes por segundo en el registro, el registro debe tener al menos:

```
logsize >= 2 * (N+1) * B
```

Es probable que el gestor de colas esté grabando la cantidad máxima de datos por segundo en el disco cuando se ejecuta en carga de trabajo máxima o puede ser cuando está grabando imágenes de soporte.

Si una transacción se ejecuta durante tanto tiempo que la extensión de registro de anotaciones que contiene el primer registro no está contenida en el registro activo, el gestor de colas retrotrae las transacciones activas de una en una, empezando por la transacción con el registro de anotaciones más antiguo.

El gestor de colas necesita dejar inactivas las extensiones de registro antiguas para que se pueda utilizar el número máximo de archivos primarios y secundarios, y el gestor de colas debe asignar otra extensión de registro.

Decida cuánto tiempo desea que la transacción de más larga ejecución se ejecute, antes de permitir que el gestor de colas la retrotraiga. La transacción de más larga ejecución puede estar esperando a que el tráfico de red sea lento o, en el caso de una transacción mal diseñada, esperando una entrada de usuario.

Puede investigar durante cuanto tiempo se ejecuta la transacción de más larga ejecución, emitiendo el siguiente mandato **runmqsc**:

```
DISPLAY CONN(*) UOWLOGDA UOWLOGTI
```

Si se emite el mandato `dspmqtrn -a`, se muestran todos los mandatos XA y no XA en todos los estados.

Al emitir este mandato se lista la fecha y hora en la que se ha grabado el primer registro de anotaciones para todas las transacciones actuales.

Atención: A efectos de cálculo del tamaño de registro, es el tiempo desde que se escribió el primer registro de anotaciones que importa, no el tiempo desde que se ha iniciado la aplicación o transacción. Redondee la longitud de la transacción de más larga ejecución al segundo más próximo. Esto se debe a las optimizaciones en el gestor de colas.

El primer registro de anotaciones puede escribirse mucho después de que se haya iniciado la aplicación, si la aplicación empieza, por ejemplo, emitiendo una llamada MQGET que esperará un tiempo antes de obtener realmente un mensaje.

Al revisar la fecha y hora máxima observada producida por el mandato

```
DISPLAY CONN(*) UOWLOGDA UOWLOGTI
```

que ha emitido originalmente desde la fecha y hora actual, puede estimar durante cuanto tiempo se ejecutará la transacción de más larga ejecución.

Asegúrese de ejecutar este mandato **runmqsc** repetidamente mientras las transacciones de más larga ejecución se están ejecutando en la carga de trabajo máxima para no subestimar la duración de la transacción de más larga ejecución.

En IBM MQ 8.0 utilice las herramientas de sistema operativo, por ejemplo **iostat** en plataformas UNIX.

Puede descubrir los bytes por segundo que el gestor de colas está grabando en el registro emitiendo el mandato siguiente:

```
amqsrua -m qmgr -c DISK -t Log
```

Los bytes lógicos escritos muestran los bytes por segundo que el gestor de colas está escribiendo en el registro. Por ejemplo:

```
$ amqsrua -m mark -c DISK -t Log
Publication received PutDate:20160920 PutTime:15383157 Interval:4 minutes,39.579 seconds
Log - bytes in use 37748736
Log - bytes max 50331648
Log file system - bytes in use 316243968
Log file system - bytes max 5368709120
Log - physical bytes written 4334030848 15501948/sec
Log - logical bytes written 3567624710 12760669/sec
Log - write latency 411 uSec
```

En este ejemplo, los bytes lógicos por segundo grabados en el registro son 12760669/sec o aproximadamente 12 MiB por segundo.

La utilización de

```
DISPLAY CONN(*) UOWLOGDA UOWLOGTI
```

mostraba que la transacción de más larga ejecución era:

```
CONN(57E14F6820700069)
EXTCONN(414D51436D61726B2020202020202020)
TYPE(CONN)
APPLTAG(msginteg_r)                     UOWLOGDA(2016-09-20)
UOWLOGTI(16.44.14)
```

Como la fecha y hora actual era 2016-09-20 16.44.19, esta transacción se había estado ejecutando durante 5 segundos. Sin embargo, requiere transacciones tolerantes que se ejecuten durante 10 segundos para que el gestor de colas pueda retrotraerlas. Por lo tanto, el tamaño de registro debe ser:

```
2 * (10 + 1) * 12 = 264 MiB
```

.

El número de archivos de registro debe poder contener el máximo tamaño de registro esperado (que se calcula en el texto anterior). Este será:

Número mínimo de archivos de registro = (Tamaño de registro necesario) / (**LogFilePages** * tamaño de página de archivo de registro (4096))

Utilizando el valor predeterminado de **LogFilePages**, que es 4096, y la estimación del tamaño de registro de 264 MiB, calculada en el texto anterior, el número mínimo de archivos de registro será:

```
264MiB / (4096 x 4096) = 16.5
```

es decir, 17 archivos de registro.

Si mide el registro de forma que la carga de trabajo esperada se ejecute en los archivos primarios:

- Los archivos secundarios proporcionan contingencias en el caso de que se requiera espacio de registro adicional.
- El registro circular siempre utiliza los archivos primarios preasignados, que es ligeramente más rápido que la asignación y desasignación de archivos secundarios.
- El gestor de colas utiliza sólo el espacio restante en los archivos primarios para calcular cuándo tomar el punto de comprobación siguiente.

Por lo tanto, en el ejemplo anterior, establezca los valores siguientes para que la carga de trabajo se ejecute en los archivos de registro primarios:

- **LogFilePages** = 4096
- **LogPrimaryFiles** = 17
- **LogSecondaryFiles** = 5

Tenga en cuenta lo siguiente:

- En este ejemplo, 5 secundarios es más del 20 por ciento del espacio de registro activo.

  El registrador intenta mantener la carga de trabajo solo en los archivos primarios. Por lo tanto, el registrador planifica puntos de comprobación cuando se llena una fracción de los archivos primarios.

  Tener los archivos secundarios es una contingencia, en el caso de que haya transacciones de larga ejecución inesperadas.

  Debe tener en cuenta que el gestor de colas toma medidas para reducir el uso de espacio de registro cuando se está utilizando más del 80 por ciento del espacio total de registro.
- Realice el mismo cálculo tanto si está utilizando el registro lineal como si utiliza el registro circular.

  No hay ninguna diferencia si está calculando el tamaño de un registro activo circular o lineal, porque el concepto del registro activo significa lo mismo en el registro lineal y en el circular.
- Las extensiones de registro necesarias para la recuperación de soporte no están dentro del registro activo y, por lo tanto, no se cuentan en el número de archivos primarios y secundarios.
- El campo *LOGUTIL* de DISPLAY QMSTATUS LOG está disponible para ayudarle a calcular, aproximadamente, el tamaño del registro activo necesario.

  Este campo está diseñado para que pueda hacer una estimación razonable del tamaño de registro necesario sin necesidad de realizar muestreos constantes para determinar la duración de las transacciones de mayor ejecución o el rendimiento máximo del gestor de colas.

¿Qué tamaño debe tener LogFilePages?

Normalmente, debe hacer que LogFilePages sea lo suficientemente grande para que se pueda aumentar fácilmente el tamaño del registro activo sin alcanzar el número máximo de archivos primarios. Es preferible tener unos pocos archivos de registro grandes que muchos archivos de registro pequeños, ya que un número reducido de archivos de registro pequeños le permite más flexibilidad para aumentar el tamaño del registro si así lo requiere.

Para el registro lineal, es posible que los archivos de registro muy grandes hagan que el rendimiento varíe. Con los archivos de registro muy grandes, hay un paso mayor para crear y dar formato a un nuevo archivo de registro, o para archivar el archivo de registro anterior. Esto es más problemático con la gestión manual de los registros de archivado, ya que con la gestión de registros automática los nuevos archivos de registro se crean automáticamente.

¿Qué sucede si mi registro es demasiado pequeño?

Puntos a tener en cuenta al estimar el tamaño mínimo del registro.

Si hace que su registro sea demasiado pequeño:

- Las transacciones de larga ejecución se restituirán.
- El siguiente punto de comprobación dese iniciarse antes de que el anterior haya finalizado.

Importante: Independientemente de la imprecisión de la estimación del tamaño del registro, se mantiene la integridad de datos.

Consulte "Utilización de la sincronización por puntos de comprobación para asegurar la recuperación completa" en la página 676 para obtener una explicación de los puntos de comprobación. Si la cantidad de espacio de registro que queda en las extensiones de registro activo se está quedando reducida, el gestor de colas planifica puntos de comprobación más frecuentemente.

Un punto de comprobación tarda cierta cantidad de tiempo; no es instantáneo. Cuantos más datos necesiten registrarse en el punto de comprobación, más tiempo tardará el punto de comprobación. Si el registro es pequeño, los puntos de comprobación se puede solapar, lo que significa que el siguiente punto de comprobación se solicita antes de que haya finalizado el punto de comprobación anterior. Si sucede esto, se escriben mensajes de error.

Si se restituyen transacciones de larga ejecución o se solapan puntos de comprobación, el gestor de colas continúa procesando la carga de trabajo. Las transacciones de vida corta continúan ejecutándose normalmente.

Sin embargo, el gestor de colas no se ejecuta de forma óptima y el rendimiento puede degradarse. Debe reiniciar el gestor de colas con suficiente espacio de registro.

¿Qué sucede si mi registro es demasiado grande?

Puntos que necesita tener en cuenta al estimar el tamaño máximo del registro.

Si hace que su registro sea demasiado grande:

- Puede aumentar el tiempo que se toma para un reinicio de emergencia, aunque esto no es probable.
- Está utilizando espacio de disco innecesario.
- Se toleran transacciones de muy larga ejecución.

Importante: Independientemente de la imprecisión de la estimación del tamaño del registro, se mantiene la integridad de datos.

Para ayudarle a calcular el tamaño máximo del registro, puede utilizar las estadísticas de utilización de registros. Para obtener más información, consulte "Cómo decidir el establecimiento de IMGLOGLN y IMGINTVL" en la página 690 y ALTER QMGR.

Consulte "Utilización de la sincronización por puntos de comprobación para asegurar la recuperación completa" en la página 676 para obtener una descripción de cómo el gestor de colas lee el registro en el reinicio. El gestor de colas reproduce el registro desde el último punto de comprobación y, a continuación, resuelve todas las transacciones que estaban activas cuando el gestor de colas ha finalizado.

Para resolver una transacción, el gestor de colas lee de nuevo todos los registros asociados con dicha transacción. Estos registros pueden poner una fecha anterior al último punto de comprobación.

Si se asigna al gestor de colas un registro muy grande, está dando al gestor de colas permiso para leer cada registro en el registro en el reinicio, aunque normalmente el gestor de colas no tiene que hacer esta tarea. Potencialmente, en el caso improbable de que esto ocurra, este proceso puede tardar mucho tiempo.

Si la sincronización por puntos de comprobación se ha detenido inesperadamente antes de que finalizara el gestor de colas, eso aumenta drásticamente el tiempo de reinicio para un gestor de colas con un registro de gran tamaño. Al limitar el tamaño del registro se limita el tiempo de reinicio de emergencia.

Para evitar estos problemas debe asegurarse de que:

- La carga de trabajo puede caber cómodamente en un registro que no es excesivamente grande.
- Evita transacciones de larga ejecución.

¿Qué tamaño debe tener el sistema de archivos de registro?

Estimación del tamaño del sistema de archivos de registro que necesita un gestor de colas.

Es importante que el sistema de archivos de registro sea lo suficientemente grande para que el gestor de colas tenga mucho espacio para escribir su registro. Si el gestor de colas llena el sistema de archivos de registro completamente, grabará FFDC, retrotraerá transacciones y puede terminar abruptamente el gestor de colas.

La cantidad de espacio de disco que se reserva para el registro debe ser al menos tan grande como el registro activo. La cantidad exacta dependerá de:

- La elección del tipo de registro (lineal o circular)
- El tamaño del registro activo (archivos primarios, archivos secundarios, páginas de archivo de registro)
- La opción de gestión de registro (manual, automática o de archivado)
- Los planes de contingencia en el caso de un objeto dañado.

Si elige un registro circular, el sistema de archivos de registro deberá ser

```
LogFilesystemSize >= (PrimaryFiles + SecondaryFiles + 1) * LogFileSize
```

Esto permite al gestor de colas grabar en todos los archivos primarios y secundarios. En casos excepcionales, el gestor de colas puede grabar una extensión adicional más allá del número de secundarios. El algoritmo anterior tiene esto en cuenta.

Si elige un registro lineal, el sistema de archivos de registro deberá ser significativamente mayor que el registro activo.

Si elige la gestión de registro manual, el gestor de colas continúa grabando en nuevas extensiones de registro según sea necesario, y es su responsabilidad suprimirlas (y archivarlas) cuando ya no sean necesarias.

El tamaño adicional del sistema de archivos de registro dependerá sobre todo de su estrategia para suprimir las extensiones superfluas o inactivas.

Puede archivar y suprimir las extensiones tan pronto como están inactivas (no se necesita una recuperación de reinicio) o puede archivar y suprimir solo las extensiones superfluas (no se necesita una recuperación de reinicio o de soportes).

Si archiva y suprime solo las extensiones superfluas y tiene un objeto dañado, **MEDIALOG** no avanzará, por lo que no habrá más extensiones superfluas. Dejará de archivar y suprimir extensiones hasta que se solucione el problema, quizás con una recuperación del objeto.

A menos que detenga la carga de trabajo, el tiempo que tiene para resolver el problema dependerá del tamaño del sistema de archivos de registro. Por lo tanto, se recomienda tener un sistema de archivos de registro generoso cuando se utiliza el registro lineal.

Si elige un registro lineal y la gestión de registro automática o de archivado, el gestor de colas reutilizará las extensiones de registro.

Las extensiones de registro que están disponibles para ser reutilizadas tienen como prefijo la letra R. Cuando se registra una imagen de soporte, a medida que se archivan las extensiones superfluas, el gestor de colas puede volver a utilizar dichas extensiones.

Por lo tanto, las extensiones reutilizadas son menores que la longitud de datos grabada en el registro entre imágenes de soporte:

```
ReuseExtents <= LogDataLengthBetweenMediaImages
```

Cuando se graban imágenes de soporte automáticamente y se establece **IMGLOGLN**, `LogDataLengthBetweenMediaImages` puede ser hasta dos veces mayor que **IMGLOGLN**, porque **IMGLOGLN** es un destino, no un máximo fijo.

Cuando se graban imágenes de soporte manualmente o se graban automáticamente en un intervalo, `LogDataLengthBetweenMediaImages` depende de la carga de trabajo y del intervalo entre las imágenes.

Además de las extensiones activas y las extensiones reutilizadas, hay extensiones inactivas (solo necesarias para la recuperación de soportes) y extensiones superfluos (no necesarias para la recuperación de reinicio o de soportes).

Cuando se utiliza la gestión de registro de archivado o automática, el gestor de colas no reutiliza las extensiones que son necesarias para la recuperación de soportes. Por lo tanto, el número de extensiones inactivas depende de la frecuencia con que se crean imágenes de soporte y de si está creándolas manual o automáticamente.

IMGINTVL y **IMGLOGLN** son destinos, no un mínimo o un máximo fijo entre imágenes de soporte. No obstante, cuando se calcula el tamaño máximo del sistema de archivos de registro necesario, es poco probable que las imágenes de soporte automáticas se graben separadas más de dos veces el valor de **IMGINTVL** o **IMGLOGLN**.

Cuando se calcula el tamaño del sistema de archivos de registro utilizando la gestión de registro automática o de archivado, también debe tener en cuenta qué puede ocurrir si se daña una cola u otro objeto. En este caso, el gestor de colas no puede crear una imagen de soporte del objeto dañado y **MEDIALOG** no avanzará.

Si la carga de trabajo continúa, el registro inactivo crecerá sin restricciones, ya que la extensión más antigua necesaria para la recuperación de soportes continúa siendo necesaria y no se puede reutilizar. Si la carga de trabajo continúa, tendrá hasta que el sistema de archivos de registro se llene completamente para solucionar el problema, antes de que el gestor de colas empiece a retrotraer transacciones e incluso termine abruptamente.

Por consiguiente, para la gestión de registro automática y de archivado:

```
LogFilesystemSize > (PrimaryFiles + SecondaryFiles +
(((TimeBetweenMediaImages *2) + TimeNeededToResolveDamagedObject) * ExtentsUsedPerHour))
* LogFilePages
```

Nota: En el algoritmo anterior se supone que se invoca **SET LOG ARCHIVED** para cada extensión, tan pronto como ya no es necesaria para la recuperación de soportes, en la gestión del registro de archivado.

Gestión de registros

El producto da soporte a la gestión automática de registros y a la recuperación automática de soportes de registros lineales. Los registros circulares son casi de autogestión, pero a veces es necesaria alguna intervención para resolver problemas de espacio.

Nota: IBM i La gestión automática y de registro de anotaciones no son válidas en IBM i.

En el registro circular, el gestor de colas solicitará espacio libre en los archivos de registro. El usuario no ve esta actividad y, normalmente, no se ve que la cantidad de espacio en disco disminuye, porque el espacio asignado se vuelve a utilizar rápidamente.

Puede suprimir archivos secundarios al utilizar el registro circular. Consulte RESET QMGR **TYPE(REDUCELOG)** para obtener más información.

En el registro lineal, el registro puede llenarse si no se ha establecido un punto de comprobación desde hace mucho tiempo, o si una transacción de ejecución larga grabó un registro de anotaciones hace mucho tiempo. El gestor de colas intenta establecer puntos de comprobación con la suficiente frecuencia como para evitar el primer problema.

Si el registro se llena, se emite el mensaje AMQ7463. Además, si las anotaciones se llenan porque una transacción de larga ejecución ha impedido que se libere el espacio, se emite el mensaje AMQ7465.

De los registros de anotaciones, sólo aquellos grabados a partir del inicio del último punto de comprobación completo, y aquellos grabados por alguna de las transacciones activas, son necesarios para reiniciar el gestor de colas.

Con el tiempo, los registros grabados más antiguos ya no serán necesarios para reiniciar el gestor de colas.

Cuando se detecta una transacción de larga ejecución, se planifica una actividad para retrotraer esa transacción. Si, por alguna razón inesperada, la retrotracción asíncrona falla, algunas llamadas MQI devuelven MQRC_RESOURCE_PROBLEM en ese caso.

Tenga en cuenta que se reserva espacio para confirmar o restituir todas las transacciones en curso, por lo que **MQCMIT** y **MQBACK** no deberían tener problemas.

Una aplicación que tiene una transacción retrotraída de este modo no puede realizar operaciones posteriores **MQPUT** o **MQGET** especificando un punto de sincronización bajo la misma transacción.

Un intento de transferir u obtener un mensaje bajo el punto de sincronización en este estado devuelve MQRC_BACKED_OUT. A continuación, la aplicación puede emitir **MQCMIT**, que devuelve MQRC_BACKED_OUT, o **MQBACK** e iniciar una nueva transacción. Cuando se ha restituido la transacción que ocupa demasiado espacio en el registro, el espacio se libera y el gestor de colas sigue funcionando normalmente.

¿Qué sucede cuando se llena un disco?

Cuando un gestor de colas está configurado para utilizar el registro lineal, el componente de registro del gestor de colas reacciona ante una condición de disco lleno de las siguientes maneras.

Si el disco que contiene los archivos de registro está lleno, entonces:

- El gestor de colas descubre esta condición solo cuando se crea un nuevo archivo de registro del tamaño requerido, y lo hace por adelantado cuando es necesario.
- Descubre la condición de disco lleno cuando el sistema operativo devuelve un error de la solicitud que indica que es necesario ampliar el archivo al tamaño requerido.
- El gestor de colas anota el mensaje AMQ6708 en el registro de errores del gestor de colas.
- Se graba un registro FFST (First Failure Support Technology) en el directorio de errores generales del sistema. Este registro proporciona detalles de la condición de disco lleno y debe conservarse por si necesita ponerse en contacto con el servicio de soporte de IBM.

Los archivos de registro se crean con un tamaño fijo, en vez de ampliarlos a medida que se van grabando registros de anotaciones. Esto significa que IBM MQ puede quedarse sin espacio de disco sólo cuando está creando un nuevo archivo; no puede quedarse sin espacio cuando está grabando un registro en el registro de anotaciones. IBM MQ sabe en todo momento la cantidad de espacio que hay disponible en los archivos de registro existentes y gestiona el espacio de los archivos en consecuencia.

Cuando utiliza el registro lineal, tiene la opción de utilizar:

- La gestión automática de extensiones de registro.

Consulte DISPLAY QMSTATUS para obtener más información sobre los nuevos atributos de registro.

Asimismo, consulte los mandatos siguientes o su equivalente PCF:

- RESET QMGR
- SET LOG para plataformas distribuidas

- Las opciones que controlan el uso de imágenes de soporte.

  Consulte el mandato ALTER QMGR y ALTER QUEUES para obtener más información sobre:

  - IMGINTVL
  - IMGLOGLN
  - IMGRCOVO
  - IMGRCOVQ
  - IMGSCHED

El registro circular devuelve un problema de recursos.

Si todavía le falta espacio, compruebe que la configuración del registro en el archivo de configuración del gestor de colas es correcta. Tal vez pueda reducir el número de archivos de registro primarios o secundarios de modo que el registro no supere el espacio disponible.

No es posible alterar el tamaño de los archivos de registro para un gestor de colas existente. El gestor de colas requiere que todas las extensiones de registro tengan el mismo tamaño.

Gestión de archivos de anotaciones

Asigne espacio suficiente para sus archivos de anotaciones. Para el registro lineal, puede suprimir archivos de registro antiguos cuando ya no los necesite.

Información específica del registro circular

Si está utilizando el registro circular, asegúrese de que hay espacio suficiente para contener los archivos de registro cuando configure el sistema (consulte "Stanza LogDefaults del archivo mqs.ini" en la página 105 y "Stanza de registro del archivo qm.ini" en la página 145). La cantidad de espacio en disco utilizada por el registro, incluido el espacio para la creación de archivos secundarios cuando es necesario, no aumenta más allá del tamaño configurado.

Información específica del registro lineal

Si está utilizando un registro lineal, los archivos de registro se añaden continuamente conforme se anotan los datos y la cantidad de espacio de disco utilizado aumenta con el tiempo. Si la velocidad a la que se anotan los datos es alta, los nuevos archivos de registro utilizan rápidamente el espacio de disco.

Con el tiempo, los archivos de registro más antiguos de un registro lineal ya no son necesarios para reiniciar el gestor de colas ni para realizar la recuperación de objetos desde soporte de ningún objeto dañado. Los métodos siguientes determinan qué archivos de registro siguen siendo necesarios:

Mensajes de sucesos del registrador de anotaciones
Cuando se produce un suceso significativo, por ejemplo, la grabación de imagen de soporte, se generan mensajes de suceso de registrador. El contenido de los mensajes de suceso de registrador especifica los archivos de registro que siguen siendo necesarios para el reinicio del gestor de colas y la recuperación desde soporte. Para obtener más información sobre los mensajes de suceso de registrador, consulte Sucesos de registrador

Estado del gestor de colas
Al ejecutar el mandato MQSC, DISPLAY QMSTATUS, o el mandato PCF, Consultar estado del gestor de colas, se devuelve información del gestor de colas, que incluye detalles de los archivos de registro necesarios. Para obtener más información sobre los mandatos MQSC, consulte Administración de IBM MQ utilizando mandatos MQSC, y para obtener información sobre los mandatos PCF, consulte Automatización de tareas de administración.

Mensajes del gestor de colas
Periódicamente, el gestor de colas emite un par de mensajes para indicar qué archivo de registro es necesario:

- El mensaje AMQ7467I proporciona el nombre del archivo de registro más antiguo necesario para reiniciar el gestor de colas. Este archivo de registro y todos los archivos de registro posteriores deben estar disponibles durante el reinicio del gestor de colas.
- El mensaje AMQ7468I proporciona el nombre del archivo de registro más antiguo necesario para la recuperación desde medio de almacenamiento.

Para determinar los archivos de registro "más antiguos" y "más recientes", utilice el número de archivo de registro en lugar de las horas de modificación aplicadas por el sistema de archivos.

Información aplicable a ambos tipos de registro

Sólo los archivos de registro necesarios para reiniciar el gestor de colas, los archivos de registro activos, deben estar en línea. Los archivos de registro inactivos se pueden copiar en un soporte de archivado, como una cinta para la recuperación tras desastre, y eliminar del directorio de registros. Los archivos de anotaciones inactivos que no son necesarios para la recuperación de objetos desde soporte se pueden considerar archivos de anotaciones superfluos. Puede suprimir archivos de registro superfluos si ya no son necesarios para el funcionamiento.

Si no se encuentra ningún archivo de registro que sea necesario, se emite el mensaje de operador AMQ6767E. Haga que el archivo de registro, y todos los archivos de registro subsiguientes, pasen a estar disponibles para el gestor de colas e intente de nuevo la operación.

Limpieza automática de extensiones de registro - solo para el registro lineal

Tiene la opción de utilizar la gestión automática de extensiones de registro lineal que ya no son necesarias para la recuperación.

Utilice el atributo **LogManagement** en la stanza de registro del archivo qm.ini, o utilizando IBM MQ Explorer para configurar la gestión automática. Consulte “Stanza de registro del archivo qm.ini” en la página 145 para obtener más información.

Consulte el parámetro LOG de **DISPLAY QMSTATUS** para obtener más detalles sobre el funcionamiento del registro, y los siguientes mandatos para utilizar el registro:

- RESET QMGR
- SET LOG

Creación automática de imágenes de soporte - solo para el registro lineal

Hay un conmutador global para controlar si el gestor de colas graba automáticamente imágenes de soporte, siendo el valor predeterminado que el conmutador no se ha establecido.

Puede controlar si se producen imágenes de soporte automáticas, y la frecuencia del proceso, utilizando los atributos de gestor de colas siguientes:

IMGSCHED
Indica si el gestor de colas graba imágenes de soporte automáticamente

IMGINTVL
La frecuencia de grabación de imágenes de soporte, en minutos

IMGLOGLN
Los megabytes de registros escritos desde la imagen de soporte anterior de un objeto.

Si tiene una hora crítica del día en la que la carga de trabajo es muy grande y desea asegurarse de que el rendimiento del sistema no se ve afectado por la creación automática de imágenes de soporte, puede desactivar temporalmente la creación automática de imágenes estableciendo **IMGSCHED***(MANUAL)*.

Puede conmutar **IMGSCHED** en cualquier momento durante la carga de trabajo.

Atención: MEDIALOG no se mueve hacia adelante si no está tomando imágenes de soporte, por lo que debe archivar las extensiones o asegurarse de que tiene suficiente espacio de disco.

También puede controlar imágenes de soporte automáticas y manuales para otros objetos definidos por el usuario utilizando el atributo **IMGRCOVO** :

- Información de autenticación
- Canal
- Conexión de cliente
- Escucha
- Lista de nombres
- Proceso
- Cola alias
- Cola local
- Servicio
- Tema

Para objetos del sistema interno como, por ejemplo, el catálogo de objetos y el objeto del gestor de colas, el gestor de colas escribe automáticamente imágenes de soporte, según sea necesario.

Consulte ALTER QMGR para obtener más información sobre los atributos.

También se pueden habilitar o inhabilitar las imágenes de soporte automáticas y manuales solo para las colas dinámicas permanentes y locales. Esto puede hacerse con el atributo de cola **IMGRCOVQ**.

Consulte ALTER QUEUE para obtener más información sobre el atributo **IMGRCOVQ** .

Notas:

1. Las imágenes de soporte solo están soportados si utiliza el registro lineal. Si ha habilitado las imágenes de soporte automáticas, pero está utilizando el registro circular, se emite un mensaje de error y está inhabilitado el atributo de imágenes de soporte automáticas del gestor de colas.
2. Si ha habilitado las imágenes de soporte automáticas, pero no ha especificado una frecuencia, ya sea de minutos o megabytes de registro, se emite un mensaje de error y no se graban imágenes de soporte automáticas.
3. Puede grabar manualmente una imagen de soporte utilizando rcdmqimg cuando haya establecido **IMGSCHED***(AUTO)*, si lo desea.

   Esto permite crear imágenes de soporte en el momento más adecuado para su empresa, por ejemplo, cuando el sistema está tranquilo. La creación de imágenes de soporte automáticas tiene en cuenta estas imágenes de soporte manuales, porque la creación de una imagen de soporte manual restablece el intervalo y la longitud del registro, antes de que se cree la siguiente la imagen de soporte automática.
4. El gestor de colas sólo graba mensajes persistentes en imágenes de soporte, no en mensajes no persistentes. Esto puede reducir el tamaño de las imágenes de soporte al migrar a versiones posteriores de IBM MQ .

Cómo decidir el establecimiento de IMGLOGLN y IMGINTVL

V 9.4.0 De forma predeterminada, **IMGLOGLN** se establece en `off` para los gestores de colas que no sean gestores de colas HA nativos. (Los gestores de colas HA nativos se crean con **IMGLOGLN** establecido en el valor del 25% del espacio disponible en el volumen donde se van a grabar los registros de recuperación.)

V 9.4.0 De forma predeterminada, **IMGINTVL** se establece en 60 minutos. El intervalo especificado por **IMGINTVL** se respeta cuando se ha llevado a cabo un trabajo nuevo suficiente en el gestor de colas para que valga la pena grabar una nueva imagen. De lo contrario, la toma de nuevas imágenes se retrasa.

Puede modificar los valores de **IMGLOGLN** y **IMGINTVL** para obtener la mejor solución para la configuración. Dé un tamaño suficiente a **IMGLOGLN** y **IMGINTVL**, para que el gestor de colas solo dedique una fracción de su tiempo a la grabación de imágenes, pero lo suficientemente pequeño para que:

- Los objetos dañados puedan recuperarse en una cantidad de tiempo razonable y
- El registro entre en el disco sin que falte espacio.

Si establece **IMGLOGLN**, una práctica recomendada es hacer que **IMGLOGLN** sea varias veces la cantidad de datos en las colas y varias veces la velocidad de datos de la carga de trabajo. Cuanto mayor sea **IMGLOGLN**, menos tiempo dedicará el gestor de colas a grabar imágenes de soporte.

De forma parecida, si establece **IMGINTVL**, una práctica recomendada es hacer que **IMGINTVL** sea varias veces la cantidad de tiempo que el gestor de colas dedica a grabar una imagen de soporte. Puede averiguar cuánto tiempo se tarda en grabar una imagen de soporte grabando una manualmente.

Si **IMGLOGLN** y **IMGINTVL** son demasiado grandes, la recuperación de un objeto dañado puede tardar mucho tiempo, porque deben reproducirse todas las extensiones desde la última imagen de soporte.

Dé un tamaño lo suficientemente pequeño a **IMGLOGLN** y **IMGINTVL**, para que el tiempo máximo dedicado a recuperar un objeto dañado sea aceptable.

Si da un tamaño muy grande a **IMGLOGLN** y **IMGINTVL**, significa que el registro crece mucho porque solo se graban imágenes de soporte excepcionalmente.

Atención: Asegúrese de que un registro de este tamaño quepa cómodamente en el sistema de archivos de registro, ya que la carga de trabajo se restituirá si el sistema de archivos de registro se llena completamente.

Puede establecer ambos, **IMGINTVL** y **IMGLOGLN**. Esto puede ser muy útil para garantizar se creen regularmente imágenes de soporte automáticas si la carga de trabajo es muy elevada (controlada por **IMGLOGLN**), pero que sigan creándose ocasionalmente cuando la carga sea muy ligera (controlada por **IMGINTVL**).

IMGINTVL y **IMGLOGLN** son los destinos del intervalo y la longitud de datos de registro entre los que se crean las imágenes de soporte automáticas.

Estos atributos no deben considerarse como un máximo o mínimo fijo. De hecho, el gestor de colas puede decidir si desea planificar una imagen de soporte automática antes, si el gestor de colas considera que es un buen momento:

- Porque la cola está vacía, por lo que crear una imagen de soporte es lo más recomendable en términos de rendimiento y
- No se ha grabado una imagen de soporte en algún tiempo

A veces, la brecha entre las imágenes de soporte automáticas puede ser un poco más grande que **IMGINTVL** o **IMGLOGLN**, o ambos.

La brecha entre las imágenes de soporte puede ser mayor que **IMGLOGLN** si la cantidad de datos en las colas se aproxima a **IMGLOGLN**. La brecha entre las imágenes de soporte puede ser mayor que **IMGINTVL** si tarda casi lo mismo que **IMGINTVL** en grabar una imagen de soporte.

Esta práctica no se recomienda porque el gestor de colas tardaría la mayor parte de su tiempo en grabar imágenes de soporte.

Cuando se utiliza la grabación de imágenes de soporte automáticas, el gestor de colas grabar una imagen de soporte para cada objeto y cola individualmente, por lo que el gestor de colas realiza un seguimiento del intervalo y la longitud de registro entre imágenes por separado para cada objeto.

Gradualmente en el tiempo, la grabación de imágenes de soporte se realiza de forma escalonada, en lugar de grabar imágenes de soporte para todos los objetos al mismo tiempo. Este escalonamiento dispersa

el impacto en el rendimiento de la grabación de imágenes de soporte y constituye otra ventaja de la grabación automática de imágenes de soporte frente a la grabación manual.

Creación manual de imágenes de soporte - solo para el registro lineal

Grabar una imagen de soporte de una cola implica escribir todos los mensajes persistentes de esa cola en el registro. Para las colas que contienen grandes volúmenes de datos de mensaje, esto implica escribir una gran cantidad de datos en el registro y este proceso puede afectar al rendimiento del sistema mientras está sucediendo.

La grabación de imágenes de soporte de otros objetos es probable que sea comparativamente rápida, puesto que la imagen de soporte de otros objetos no contiene datos de usuario.

Debe considerar atentamente cuándo se deben grabar las imágenes de soporte de las colas, para que el proceso no interfiera con la carga de trabajo máxima.

Debe grabar la imagen de soporte de todos los objetos con regularidad, a fin de actualizar la extensión de registro más antigua necesaria para la recuperación de soporte.

Lo más apropiado es grabar la imagen de soporte de una cola es cuando está vacía, porque en ese momento no se escriben datos de mensaje en el registro. Y a la inversa, el peor momento es cuando la cola es muy profunda o contiene mensajes muy grandes.

Un buen momento para grabar la imagen de soporte de una cola es cuando el sistema está tranquilo; mientras que un mal momento es durante la carga de trabajo máxima. Si la carga de trabajo es siempre tranquila a medianoche, por ejemplo, puede decidir grabar las imágenes de soporte cada día a medianoche.

Escalonar la grabación de cada una de las colas puede dispersar el impacto en el rendimiento y, así, reducir su efecto. Cuanto más tiempo haya pasado desde que ha grabado imágenes de soporte por última vez, más importante resulta grabarlas, porque el número de extensiones de registro necesarias para la recuperación de soporte está aumentando.

Nota: Al realizar la recuperación desde soporte, todos los archivos de registro necesarios deben estar disponibles en el directorio del archivo de registro a la vez. Asegúrese de captar imágenes de soporte con regularidad de todos los objetos que desee recuperar y así no se quedará sin espacio de disco para conservar todos los archivos de registro que necesite.

Por ejemplo, para crear una imagen de soporte de todos los objetos del gestor de colas, ejecute el mandato **rcdmqimg**, tal como se muestra en los ejemplos siguientes:

En Windows

```
rcdmqimg -m QMNAME -t all *
```

En AIX and Linux

```
rcdmqimg -m QMNAME -t all "*"
```

Ejecutar **rcdmqimg** hace que avance el número de secuencia de registro cronológico (LSN) del soporte. Para obtener más detalles sobre los números de secuencia de registro cronológico, consulte “Volcado del contenido del registro mediante el mandato dmpmqlog” en la página 702. **rcdmqimg** no se ejecuta automáticamente y, por tanto, se debe ejecutar manualmente o desde una tarea automática que haya creado. Si desea más información sobre este mandato, consulte rcdmqimg y dmpmqlog.

El registro manual de imágenes de soporte con **rcdmqimg** para gestionar espacio de registro no es necesario, si ha optado por utilizar el registro lineal con la toma automática de imágenes de soporte controlada por el gestor de colas.

Nota: Los mensajes AMQ7467 y AMQ7468 también se pueden emitir en el momento en que se ejecuta el mandato `rcdmqimg`.

Imágenes de soporte parciales

Se recomienda utilizar los mensajes de IBM MQ solo para los datos que se espera que se van a consumir en un futuro próximo, para que cada mensaje esté en una cola durante una cantidad de tiempo relativamente breve.

Por el contrario, no se recomienda utilizar mensajes de IBM MQ para almacenar datos a largo plazo como, por ejemplo, para una base de datos.

También se recomienda garantizar que las colas sean relativamente superficiales, pero no tener colas superficiales cuyos mensajes hayan estado en la cola mucho tiempo.

Si sigue estas directrices, permite que el gestor de colas optimice el rendimiento del registro automático de imágenes de soporte.

La grabación de la imagen de soporte de una cola vacía es muy productivo (desde un punto de vista del rendimiento), mientras que la creación de una imagen de soporte de una cola con una gran cantidad de datos no es nada productivo, porque todos esos datos deben grabarse en el registro en la imagen de soporte.

Para las colas superficiales con mensajes transferidos recientemente, el gestor de colas puede realizar una optimización adicional.

Si todos los mensajes que hay actualmente en la cola se han colocado recientemente, el gestor de colas pueda grabar la imagen de soporte como si estuviera en un momento (*punto de recuperación*) justo antes de que se hayan colocado todos los mensajes, para poder así grabar la imagen de la cola vacía. Este proceso tiene un coste muy bajo en términos de rendimiento.

Si todos los mensajes que estaban en la cola en el punto de recuperación se han obtenido posteriormente, no es necesario que estos mensajes se registren en la imagen de soporte, ya que ya no están en la cola.

Esto se denomina una *imagen de soporte parcial*. Posteriormente, en el caso improbable de que la cola deba recuperarse, se reproducirán todos los registros relacionados con esta cola desde la última imagen de soporte, para restaurar así todos los mensajes colocados recientemente.

Aumente hubiera pocos mensajes en la cola en el punto de recuperación que estén actualmente en la cola (y que, por lo tanto, deben grabarse en la imagen de soporte parcial), continúa siendo más productivo grabar esta imagen de soporte parcial que una imagen de soporte completa de todos los mensajes.

Asegurarse de que los mensajes permanezcan en las colas un breve periodo de tiempo probablemente aumentará el rendimiento de la grabación automática de imágenes de soporte.

Ubicación del archivo de registro
Al elegir una ubicación para sus archivos de registro, recuerde que el funcionamiento general se verá gravemente afectado si IBM MQ no da formato a un nuevo archivo de registro por falta de espacio de disco.

Si está utilizando un registro circular, asegúrese de que hay espacio suficiente en la unidad para al menos los archivos de registro primarios configurados. También debe dejar espacio para al menos un archivo de registro secundario, que será necesario si el registro tiene que crecer.

Si está utilizando un registro lineal, deberá dejar mucho más espacio; el espacio que consume el registro aumenta continuamente a medida que se añaden datos.

Debe colocar los archivos de registro en otra unidad de disco que no sea la de los datos del gestor de colas.

La integridad de los datos en este dispositivo es primordial; debe permitir la redundancia incorporada.

Puede que también sea posible colocar los archivos de registro en varias unidades de discos duplicadas. Esto le protegerá en caso de que la unidad que contiene las anotaciones sufra una anomalía. Sin la duplicación de discos, podría verse forzado a recurrir a la última copia de seguridad del sistema IBM MQ.

ALW *Cambio del registro del gestor de colas de lineal a circular*

Puede cambiar el tipo de registro del gestor de colas de lineal a circular utilizando el mandato **migmqlog** .

Antes de empezar

Revise los Tipos de registro para decidir si desea utilizar el registro lineal o circular.

Decida si desea cambiar el tipo de registro en su lugar o mueva el registro a una nueva ubicación. Cuando mueve el registro a una nueva ubicación utilizando el mandato **migmqlog** , la vía de acceso del registro en el archivo qm.ini se actualiza para que cuando inicie el gestor de colas utilice el registro modificado. Puede especificar una nueva ubicación utilizando la opción **-ld** . Si cambia el gestor de colas de un disco antiguo a un disco nuevo de formato avanzado, puede ser conveniente utilizar la opción **-ld**.

Procedimiento

1. Inicie una sesión como miembro del grupo `mqm`.
2. Asegúrese de que tiene suficiente espacio para cambiar el registro. Debe asegurarse de que haya espacio para al menos los archivos de registro primarios configurados y un archivo de registro secundario.
3. Si todavía no lo ha hecho, detenga el gestor de colas utilizando el mandato **endmqm -w** .
4. Si todavía no lo ha hecho, realice una copia de seguridad del gestor de colas.

   Para obtener más información, consulte los apartados “Hacer copia de seguridad de los datos de gestor de colas” en la página 707 y “Copia de seguridad de los archivos de configuración después de crear un gestor de colas” en la página 14.
5. Ejecute el mandato **migmqlog**:

   - Si elige cambiar el tipo de registro sin cambiar la ubicación del registro, utilice el mandato siguiente:

     ```
     migmqlog -m QMgrName -lc
     ```

   - Si elige cambiar el tipo de registro y mover el registro a una nueva ubicación, utilice el mandato siguiente:

     ```
     migmqlog -m QMgrName -lc -ld NewLogLocation
     ```

     donde *Ubicación deNewLog* es una vía de acceso de archivo absoluta que especifica la nueva ubicación del archivo de registro. No utilice una vía de acceso de archivo relativa con el parámetro **-ld** .

   Para obtener más información, consulte **migmqlog**.

   Si, por alguna razón, por ejemplo, debido a una interrupción de la alimentación, el mandato **migmqlog** se detiene antes de que haya completado el proceso, vuelva a ejecutar el mismo mandato **migmqlog** en los registros modificados parcialmente para completar los cambios.

Resultados

El mandato se ejecuta y el tipo de registro del gestor de colas se actualiza. Tenga en cuenta que **migmqlog** puede tardar unos minutos en completarse si el registro es muy grande. No obstante, el mandato genera mensajes de progreso de vez en cuando.

Tareas relacionadas

“Cambio del registro del gestor de colas de circular a lineal” en la página 694

Puede cambiar el tipo de registro del gestor de colas de circular a lineal utilizando el mandato **migmqlog** .

ALW *Cambio del registro del gestor de colas de circular a lineal*

Puede cambiar el tipo de registro del gestor de colas de circular a lineal utilizando el mandato **migmqlog** .

Antes de empezar

Revise los Tipos de registro para decidir si desea utilizar el registro lineal o circular.

Decida si desea cambiar el tipo de registro en su lugar o mueva el registro a una nueva ubicación. Cuando mueve el registro a una nueva ubicación utilizando el mandato **migmqlog** , la vía de acceso del registro en el archivo qm.ini se actualiza para que cuando inicie el gestor de colas utilice el registro modificado. Puede especificar una nueva ubicación utilizando la opción **-ld** . Si cambia el gestor de colas de un disco antiguo a un disco nuevo de formato avanzado, puede ser conveniente utilizar la opción **-ld**.

Acerca de esta tarea

Atención: Después de haber cambiado el registro, no se habrá registrado una imagen de soporte cuando se inicie el gestor de colas. Planifique cómo desea grabar las imágenes de soporte, ya sea automáticamente estableciendo los atributos:

- IMGSCHED
- IMGINTVL
- IMGLOGLN
- IMGRCOVO
- IMGRCOVQ

en ALTER QMGR, o manualmente ejecutando de forma periódica **rcdmqimg**.

Procedimiento

1. Inicie una sesión como miembro del grupo `mqm`.
2. Asegúrese de que tiene suficiente espacio para cambiar el registro. El espacio utilizado por un registro lineal aumenta continuamente a medida que se registran los datos.
3. Si todavía no lo ha hecho, detenga el gestor de colas utilizando el mandato **endmqm -w** .
4. Si todavía no lo ha hecho, realice una copia de seguridad del gestor de colas.

   Para obtener más información, consulte los apartados “Hacer copia de seguridad de los datos de gestor de colas” en la página 707 y “Copia de seguridad de los archivos de configuración después de crear un gestor de colas” en la página 14.
5. Ejecute el mandato **migmqlog**. Tenga en cuenta que **migmqlog** puede tardar unos minutos en completarse si el registro es muy grande. No obstante, el mandato genera mensajes de progreso de vez en cuando.

   - Si elige cambiar el tipo de registro sin cambiar la ubicación del registro, utilice el mandato siguiente:

     ```
     migmqlog -m QMgrName -ll
     ```

   - Si elige cambiar el tipo de registro y mover el registro a una nueva ubicación, utilice el mandato siguiente:

     ```
     migmqlog -m QMgrName -ll -ld NewLogLocation
     ```

     donde *Ubicación deNewLog* es una vía de acceso de archivo absoluta que especifica la nueva ubicación del archivo de registro. No utilice una vía de acceso de archivo relativa con el parámetro **-ld** .

   Para obtener más información, consulte **migmqlog**.

   Si, por alguna razón, por ejemplo, debido a una interrupción de la alimentación, el mandato **migmqlog** se detiene antes de que haya completado el proceso, vuelva a ejecutar el mismo mandato **migmqlog** en los registros modificados parcialmente para completar los cambios.
6. Inicie el gestor de colas y establezca los atributos de cola y recuperación imagen correspondientes para su entorno.

7. Considere cuándo se deben grabar las imágenes manuales de los objetos recuperables.

Tareas relacionadas

“Cambio del registro del gestor de colas de lineal a circular” en la página 694
Puede cambiar el tipo de registro del gestor de colas de lineal a circular utilizando el mandato **migmqlog** .

Determinación de los archivos de registro superfluos - solo para registro lineal
En el registro circular, no suprima nunca datos del directorio de registro. Al gestionar archivos de registro lineal, es importante asegurarse de qué archivos se pueden suprimir o archivar. Esta información le ayudará a tomar esta decisión.

No utilice las horas de modificación del sistema de archivos para determinar los archivos de registro "más antiguos". Utilice sólo el número de archivo de anotaciones. El uso de los archivos de registro por parte del gestor de colas sigue reglas complejas, incluyendo la preasignación y formato de los archivos de registro antes de que se necesiten. Puede ver archivos de registro con horas de modificación que le llevarán a conclusiones erróneas si intenta utilizar estas horas para determinar la antigüedad relativa.

Para determinar el archivo de registro más antiguo que se necesita, puede utilizar tres opciones:

- El mandato DISPLAY QMSTATUS
- Los mensajes de suceso de registrador y, por último,
- Los mensajes de registro de errores

Para el mandato DISPLAY QMSTATUS, para determinar la extensión de registro más antigua necesaria para:

- Reiniciar el gestor de colas, emita el mandato `DISPLAY QMSTATUS RECLOG`.
- Ejecutar una recuperación de soportes, emita el mandato `DISPLAY QMSTATUS MEDIALOG`.
- Determine el nombre de la notificación de archivado, emita el mandato `DISPLAY QMSTATUS ARCHLOG`.

Puede reducir el número de extensiones de registro secundario cuando se utiliza el registro circular emitiendo el mandato **RESET QMGR TYPE(REDUCELOG)**.

En general, un número de archivo de registro inferior implica un registro más antiguo. A menos que tenga un volumen de archivos de registro muy alto, del orden de 3000 archivos de registro por día durante 10 años, no necesita contemplar la posibilidad de que el número se reinicie en 9.999.999. En este caso, puede archivar cualquier archivo de registro con un número que sea inferior al valor RECLOG, y puede suprimir cualquier archivo de registro con un número que sea inferior a los valores RECLOG y MEDIALOG.

Atención: El archivo de registro se reinicia, por lo que el número siguiente después de 9999999 es cero.

Inicio en frío: ¿qué hacer si las extensiones de registro están dañadas o faltan?

Si la empresa pierde algunas o todas las extensiones de registro necesarias para la recuperación del reinicio, el gestor de colas no podrá reproducir el registro de recuperación y, por lo tanto, no se podrá reiniciar. Si requiere que el gestor de colas se reinicie cuando el registro de recuperación esté dañado de alguna forma, a expensas de mantener la integridad de datos, es posible hacerlo, aunque se desaconseja con firmeza. Este proceso se conoce como *inicio en frío* de un gestor de colas.

Importante: Sólo se debe considerar la posibilidad de realizar un inicio en frío de un gestor de colas en circunstancias excepcionales, ya que implica riesgos para la integridad de los datos, que se describen en esta página. IBM sugiere que vuelva a crear un gestor de colas, preferentemente al inicio en frío, en respuesta a archivos de datos dañados.

Si se requiere un inicio en frío por razones operativas, póngase en contacto con su representante de soporte de IBM para revisar la causa raíz del problema. Debe reemplazar lo antes posible un gestor de colas reiniciado en frío con un gestor de colas que se haya vuelto a crear.

Los efectos del inicio en frío

En el inicio en frío, el gestor de colas crea un registro de recuperación vacío y se basa en los datos de los archivos de cola y otros archivos de objeto en su estado existente. Puesto que los datos de los archivos de cola pueden ser incoherentes, los mensajes se podrían perder, duplicar, dañar o ser incoherentes.

El gestor de colas almacena la configuración de todos los demás objetos persistidos en el registro de recuperación, así como en archivos de objeto. También se registran otros datos de estado interno en el registro de recuperación, así que en el inicio en frío, los datos de estado interno se restablecen y todos los demás datos de configuración podrían ser inexactos.

Los efectos del inicio en frío son impredecibles y de gran alcance, por lo que debería evitar un inicio en frío, a menos que sea absolutamente necesario. Después del inicio en frío, la información de los archivos de cola y objeto pueden ser tan incoherentes que el gestor de colas no se reiniciará.

Si el gestor de colas no se reinicia, no hay una forma simple de descubrir en qué datos de mensaje o configuración se puede basar y cuáles no. Asimismo, después de un inicio en frío, las colas podrían estar dañadas y, por lo tanto, volverse totalmente inutilizables.

Además, si obtiene de, o coloca en, una cola concreta, los mensajes de dicha cola podrían estar dañados, podrían faltar o estar duplicados. Las transacciones y los canales pueden estar pendientes de duda. Aunque el gestor de colas se inicie en frío correctamente y las colas parezcan intactas, los efectos impredecibles del inicio en frío podrían no hacerse evidentes hasta mucho más tarde.

Qué hacer si necesita hacer un inicio en frío

La realización de un inicio en frío no se debe considerar una práctica operativa estándar, e IBM recomienda encarecidamente que no lo haga. Sin embargo, si se encuentra en una posición donde necesita definitivamente iniciar en frío un gestor de colas, póngase en contacto con IBM MQ Support.

El proceso de inicio en frío de un gestor de colas solía ser mucho más complejo para un gestor de colas lineal que uno circular. En IBM MQ 9.1.3, el proceso de inicio en frío se ha simplificado mucho, y ya no implica la copia o el cambio de nombre de las extensiones de registro.

Póngase en contacto con el soporte de IBM , que le proporcionará una clave que pasará al mandato **strmqm** para iniciar en frío un gestor de colas.

Atención: El mandato coldstart todavía conlleva los mismos riesgos de perder la integridad de los datos que un inicio en frío manual, y IBM le recomienda encarecidamente que no lo haga.

Eliminación de futuros inicios en frío: una solicitud

El mandato strmqm requiere una clave para iniciar en frío, porque IBM MQ desea que se ponga en contacto con el equipo de soporte de IBM MQ, si necesita realizar un inicio en frío, ya que IBM MQ está deseoso de comprender cómo ha acabado en esta situación.

El inicio en frío es claramente algo que es mejor evitar. IBM MQ ha realizado un considerable esfuerzo para asegurarse de que no tendrá que iniciar en frío el gestor de colas, y IBM desea descubrir si hay algo más que pueda realizar el producto para mitigar el hecho de tener que iniciar en frío.

Precauciones para evitar un inicio en frío

El método de registro predeterminado al crear un gestor de colas es el registro circular. Con el registro circular, otorga al gestor de colas un número particular de extensiones de registro primario y secundario de un tamaño determinado. Cree el sistema de archivos del registro lo suficientemente grande como para contener todas las extensiones de registro primario y secundario, y nunca debería tener que administrarlas.

De forma alternativa, puede utilizar el registro lineal en oposición al circular. El registro lineal le proporciona la capacidad añadida de recuperar colas y otros objetos, en el caso improbable de que se haya dañado. Pero, de forma predeterminada, el registro lineal requiere que suprima las extensiones de

registro que ya no son necesarias para el reinicio o la recuperación de soporte. Se conoce a este proceso como gestión de registro manual.

Cuando se administran las extensiones de registro de esta forma, es posible suprimir de forma inadvertida demasiadas extensiones de registro y, por lo tanto, terminar teniendo que realizar un inicio en frío. Para mitigar este riesgo, utilice la gestión de registros automático, de modo que el gestor de colas gestiona las extensiones de registro en su nombre.

La práctica recomendada es colocar el registro de recuperación en un sistema de archivos de registro separado que solo contiene el registro de recuperación. Si coloca el registro de recuperación en el mismo sistema de archivos que el resto del gestor de colas, a veces, puede descubrir que el sistema de archivos se está llenando de forma accidental debido, quizás, a archivos de cola grandes. Convierta el directorio de registro para el gestor de colas en un sistema de archivos separado, o especifique un sistema de archivos de registro diferente utilizando la opción de línea de mandatos **-ld** en el mandato **crtmqm**.

Si el sistema de archivos que contiene los archivos de cola se llena, es posible que no pueda colocar nada en estas colas, pero el gestor de colas se sigue ejecutando. Si el sistema de archivos que contiene el registro de recuperación se llena, el gestor de colas finaliza de forma abrupta y no se reiniciará hasta que libere algún espacio.

Tenga cuidado en no suprimir las extensiones de registro necesarias para la recuperación del reinicio, de lo contrario, es posible que tenga que realizar un inicio en frío. A veces, es posible que descubra que necesita realizar un inicio en frío porque el disco que ha fallado contiene su registro de recuperación. El procedimiento recomendado es colocar el registro de recuperación en un disco replicado y, así, mitigar el riesgo de una anomalía de disco.

El traslado de los mensajes y la configuración a un nuevo gestor de colas de sustitución evita la posibilidad de problemas continuados con un gestor de colas que se ha iniciado en frío previamente.

Conserve una nota sobre qué gestores de colas se han inicio en frío previamente, aunque se hayan inicio en frío hace mucho tiempo y se hayan detenido, reiniciado y migrado mientras tanto. Cuando se ponga en contacto con el equipo de soporte de IBM, diga si el gestor de colas se ha iniciado en frío previamente y, si es así, proporcione la máxima información posible sobre qué ha provocado el requisito de un inicio en frío.

Utilización del registro para la recuperación

Puede utilizar la información de los registros como ayuda para la recuperación de anomalías.

Los datos pueden quedar dañados por diversos motivos. IBM MQ le ayuda a recuperarse de lo siguiente:

- Un objeto de datos dañado
- Una pérdida de alimentación en el sistema
- Una anomalía en las comunicaciones

En esta sección se contempla cómo se pueden utilizan los registros para recuperarse de estos problemas.

Recuperación de pérdida de alimentación o de anomalías de comunicaciones

IBM MQ puede recuperarse de anomalías de comunicaciones y de una pérdida de alimentación. Además, a veces es posible recuperarse de otros tipos de problemas, tales como la supresión accidental de un archivo.

En el caso de que se produzca una anomalía en las comunicaciones, los mensajes persistentes permanecen en las colas hasta que una aplicación receptora los elimina. Si se va a transmitir el mensaje, permanecerá en la cola de transmisión hasta que se pueda transmitir satisfactoriamente. Para recuperarse de una anomalía en las comunicaciones, generalmente es suficiente con reiniciar los canales utilizando el enlace que ha fallado.

Si sufre una pérdida de alimentación, cuando el gestor de colas se reinicia, IBM MQ restaurar las colas al estado de confirmación que tenían cuando se produjo la anomalía. Esto asegura que no se pierda ningún mensaje persistente. Los mensajes no persistentes se descartan; no perduran cuando IBM MQ se detiene repentinamente.

Recuperación de objetos dañados

Un objeto de IBM MQ puede quedar inutilizable de varias maneras, por ejemplo, debido a que se ha dañado accidentalmente. Entonces, deberá recuperar la totalidad del sistema o parte del mismo. La acción necesaria depende del momento en que se detecta el daño, de si el método de registro seleccionado da soporte a la recuperación de objetos desde soporte y de los objetos que estén dañados.

Recuperación desde medio de almacenamiento

Puede grabar imágenes de soporte para objetos de forma que se puedan recuperar si están dañados. Esta característica sólo está disponible en los gestores de colas que utilizan el registro lineal o el registro replicado y, para el registro lineal, sólo para los objetos definidos como recuperables. Puede definir que los tipos de objeto son recuperables utilizando los atributos de gestor de colas **IMGRCOVO** y **IMGRCOVQ** , consulte ALTER QMGR. Si un objeto que no está definido como recuperable está dañado, las opciones de recuperación son las mismas que para el registro circular.

La recuperación de soporte vuelve a crear objetos a partir de la información registrada en un registro lineal o en un registro replicado. Por ejemplo, si el archivo de un objeto se suprime accidentalmente, o queda inutilizable por cualquier otro motivo, se puede utilizar la recuperación desde soporte para volver a crearlo. La información del registro necesaria para la recuperación desde soporte se denomina *imagen de soporte*.

Una imagen de soporte es una secuencia de registros de anotaciones que contienen una imagen de un objeto a partir de la cual se puede volver a crear dicho objeto.

El primer registro necesario para volver a crear un objeto se conoce como su *registro de recuperación de objetos desde soporte*; se trata del principio de la imagen de soporte más reciente del objeto. El registro de recuperación desde soporte de cada objeto es uno de los fragmentos de información registrados durante un punto de comprobación.

Cuando se vuelve a crear un objeto a partir de su imagen de soporte, también es necesario volver a ejecutar todos los registros de anotaciones que describen las actualizaciones realizadas en el objeto desde que se llevó a cabo la última imagen.

Por ejemplo, piense en una cola local que tiene una imagen del objeto de cola tomada para que se transfiriera a la cola un mensaje persistente. Para volver a crear la imagen más reciente del objeto, es necesario reproducir las entradas de registro que registran la transferencia del mensaje a la cola, además de reproducir la propia imagen.

Cuando se crea un objeto, los registros de anotaciones que se graban contienen suficiente información para volver a crear por completo el objeto. Estos registros forman la primera imagen de soporte del objeto. A continuación, cada vez que se concluye, el gestor de colas registra automáticamente las imágenes de soporte como se indica a continuación:

- Las imágenes de todos los objetos de proceso y de colas que no son locales
- Las imágenes de las colas locales vacías

Las imágenes de soporte también se pueden registrar manualmente mediante el mandato **rcdmqimg**, que se describe en rcdmqimg. Este mandato graba una imagen de soporte del objeto de IBM MQ.

El gestor de colas graba automáticamente las imágenes de soporte si se establece **IMGSCHED***(AUTO)*. Para obtener más información, consulte ALTER QMGR para obtener información sobre **IMGINTVL** y **INGLOGLN**.

Cuando se ha grabado una imagen de soporte, para volver a crear objetos dañados, sólo serán necesarios los registros que contienen la imagen de soporte y todos los registros creados posteriormente. Las ventajas de crear imágenes de soporte dependen de factores como la cantidad de almacenamiento libre disponible y la velocidad a la que se crean los archivos de registro.

Recuperación desde imágenes de soporte

Un gestor de colas recupera automáticamente algunos objetos de su imagen de soporte durante el inicio del gestor de colas. Recupera una cola automáticamente si estaba implicada en cualquier transacción que

estaba incompleta cuando el gestor de colas se cerró por última vez, y se dañó durante el proceso de reinicio.

Debe recuperar manualmente otros objetos, mediante el mandato **rcrmqobj**, que reproduce los registros de las anotaciones para volver a crear el objeto de IBM MQ. El objeto se vuelve a crear a partir de la imagen más reciente que hay en el registro, junto con todos los sucesos de anotaciones aplicables generados entre el momento en que se guardó la imagen y el momento en que se emitió el mandato para volver a crearlo. Si un objeto de IBM MQ queda dañado, las únicas acciones válidas que se pueden realizar son suprimirlo o volver a crearlo con este método. Los mensajes que no son persistentes no se pueden recuperar de este modo.

Consulte rcrmqobj para ver más detalles del mandato **rcrmqobj**.

El archivo de registro que contiene el registro de recuperación desde soporte, y todos los archivos de registro subsiguientes, deben estar disponibles en el directorio de archivos de registro cuando intente recuperar un objeto desde soporte. Si no se puede encontrar un archivo necesario, se emite el mensaje de operador AMQ6767 y la operación de recuperación de objetos desde soporte no se realiza satisfactoriamente. Si no capta regularmente imágenes de soporte de los objetos que desea volver a crear, se puede encontrar con que no tiene espacio de disco suficiente para todos los archivos de registro necesarios para volver a crear un objeto.

OpenShift V 9.4.0 Los gestores de colas HA nativos utilizan el registro replicado. Estos gestores de colas intentan la recuperación automática de objetos elegibles cuando se detecta un daño. Una vez iniciado, los gestores de colas de HA nativa, de forma predeterminada, intentan automáticamente la recuperación asíncrona cuando se detecta un daño en el objeto. Es posible que la recuperación no sea posible inmediatamente si, por ejemplo, una aplicación está utilizando el objeto, o las extensiones de registro necesarias para la recuperación desde soporte no están disponibles. En estas situaciones, el proceso de recuperación asíncrona se reintenta periódicamente. Si se resuelve el problema que ha impedido la recuperación, el objeto se recuperará en el siguiente reintento, o el objeto se puede recuperar manualmente, utilizando el mandato **rcrmqobj** .

Qué objetos de archivo existen

El gestor de colas almacena los atributos de objetos que se definen en **runmqsc** en archivos en disco. Estos archivos de objeto están en subdirectorios bajo el directorio de datos del gestor de colas.

Linux AIX Por ejemplo, en plataformas AIX and Linux, los canales se almacenan en `/var/mqm/qmgrs/qmgr/channel`.

Los datos de estos archivos de objeto son imagen de soporte de los objetos. Si estos archivos de objeto se suprimen o corrompen, se daña el objeto almacenado en ese archivo. Mediante el uso de un gestor de colas de registro lineal, los objetos dañados pueden recuperarse del registro mediante el mandato rcrmqobj. Los gestores de colas de registro replicado (HA nativa) intentan automáticamente recuperar los objetos dañados cuando se detectan.

La mayoría de los archivos de objeto contienen sólo los atributos del objeto, de modo los archivos de canal contienen los atributos de canales. Existen las siguientes excepciones:

- Catálogo

  El catálogo de objetos cataloga todos los objetos de todos los tipos y se almacena en `qmanager/QMQMOBJCAT`.
- Archivos de sincronización

  El archivo de sincronización contiene datos de estado internos asociados con todos los canales.
- Colas

  Los archivos de cola contienen los mensajes en esa cola, así como los atributos de dicha cola.

Tenga en cuenta que no hay ningún catálogo u objeto de archivo de sincronización expuesto en **runmqsc** o IBM MQ Explorer.

El catálogo y el gestor de colas pueden registrarse, pero no recuperarse. Si estos objetos se dañan, el gestor de colas finaliza de forma preventiva y estos objetos se recuperan automáticamente en el reinicio.

Las suscripciones no se listan en objetos a registrar o recuperar, porque las suscripciones duraderas se almacenan en una cola de sistema. Para registrar o recuperar suscripciones duraderas, registre o recupere SYSTEM.DURABLE.SUBSCRIBER.QUEUE en su lugar.

Recuperación de objetos dañados durante el inicio

Si el gestor de colas detecta un objeto dañado durante el inicio, la acción que tome dependerá del tipo de objeto y de si el gestor de colas está configurado para realizar la recuperación de objetos desde soporte.

Si el objeto de gestor de colas está dañado, el gestor de colas no podrá iniciarse si no puede recuperar el objeto. Si el gestor de colas está configurado con un registro lineal y, por tanto, da soporte a la recuperación desde soporte, IBM MQ intenta automáticamente volver a crear el objeto de gestor de colas a partir de sus imágenes de soporte. Si el método de registro seleccionado no da soporte a la recuperación desde soporte, puede restaurar una copia de seguridad del gestor de colas o suprimir el gestor de colas.

Si había transacciones activas cuando se detuvo el gestor de colas, las colas locales que contienen los mensajes persistentes no confirmados, transferidos u obtenidos dentro de estas transacciones, también son necesarias para iniciar el gestor de colas satisfactoriamente. Si alguna de estas colas locales está dañada y el gestor de colas da soporte a la recuperación de objetos desde soporte, intenta automáticamente volver a crearla a partir de sus imágenes de soporte. Si no se puede recuperar alguna de las colas, IBM MQ no puede iniciarse.

Si durante el proceso de inicio se detecta alguna cola local dañada que contiene mensajes no confirmados de un gestor de colas que no permite la recuperación de objetos desde soporte, la cola se marca como objeto dañado y se ignoran los mensajes no confirmados que contiene. Esta situación se debe a que no es posible realizar la recuperación desde soporte de objetos dañados en un gestor de colas de estas características y la única acción posible es suprimirlos. Se emite el mensaje AMQ7472 para indicar los daños.

Recuperación de objetos dañados en otras ocasiones

La recuperación de medios de objetos sólo es automática durante el inicio (que no sea para gestores de colas de HA nativa, que utilizan la recuperación automática de forma predeterminada). En otras ocasiones, cuando se detecta un daño en el objeto, se emite el mensaje de operador AMQ7472 y la mayoría de las operaciones que utilizan el objeto fallan con el código de retorno MQRC_OBJECT_DAMAGED. Si el objeto del gestor de colas está dañado en algún momento después de que se haya iniciado el gestor de colas, el gestor de colas realiza una conclusión preferente. Cuando un objeto ha quedado dañado puede suprimirlo o, si el gestor de colas utiliza una anotación lineal, intentar recuperarlo a partir de su imagen de soporte mediante el mandato **rcrmqobj** (consulte rcrmqobj para ver detalles adicionales).

Si una cola (u otro objeto) se daña, **MEDIALOG** no avanzará. Esto se debe a que **MEDIALOG** es la extensión más antigua necesaria para la recuperación de soporte. Si la carga de trabajo es continua, **CURRLOG** seguirá avanzando y se grabarán extensiones nuevas. En función de la configuración (incluyendo su valor **LogManagement**), esto podría empezar a rellenar el sistema de archivos del registro. Si el sistema de archivos de registro se llena completamente, las transacciones se retrotraen y el gestor de colas puede terminar abruptamente. Por lo tanto, cuando se daña una cola, es posible que solo tenga una cantidad limitada de tiempo para actuar antes de que finalice el gestor de colas. La cantidad de tiempo dependerá de la velocidad con la que la carga de trabajo esté haciendo que el gestor de colas grabe nuevas extensiones y de la cantidad de espacio libre que tenga en el sistema de archivos de registro.

Si utiliza la gestión de registro manual, puede que esté archivando extensiones que no sean necesarias para la recuperación de reinicio y suprimiéndolas del sistema de archivos de registro, aunque sigan siendo necesarias para la recuperación de soporte. Esto es aceptable siempre que pueda restaurarlas desde el archivado cuando sea necesario. Esta política no hace que se llene el sistema de archivos de registro cuando se daña una cola y **MEDIALOG** deja de avanzar. Sin embargo, si solo archiva y suprime

extensiones que no son necesarios para un reinicio o una recuperación de soporte, el sistema de archivos del registro empieza a llenarse si se daña una cola.

Si utiliza la gestión de registro de archivado o automática, el gestor de colas no reutilizará las extensiones que siguen siendo necesarias para la recuperación de soporte, aunque las haya archivado y haya notificado al gestor de colas utilizando SET LOG ARCHIVED. Por lo tanto, si se daña una cola, el sistema de archivos de registro empezará a llenarse.

Si se daña una cola, se grabarán FFDC de OBJECT DAMAGED y **MEDIALOG** dejará de avanzar. El objeto dañado puede identificarse a partir del FFDC o porque es el objeto con el **MEDIALOG** más antiguo cuando visualiza su estado en **runmqsc**.

Si el sistema de archivos de registro se está llenado y está preocupado de que la carga de trabajo se restituya porque el sistema de archivos de registro ya no tenga espacio, se recomienda recuperar el objeto o desactivar temporalmente la carga de trabajo.

En el caso de los gestores de colas de HA nativa, que utilizan el registro replicado, se intenta la recuperación automática de objetos dañados. Una vez iniciado, los gestores de colas de HA nativa, de forma predeterminada, intentan automáticamente la recuperación asíncrona cuando se detecta un daño en el objeto. Es posible que la recuperación no sea posible inmediatamente si, por ejemplo, una aplicación está utilizando el objeto, o las extensiones de registro necesarias para la recuperación desde soporte no están disponibles. En estas situaciones, el proceso de recuperación asíncrona se reintenta periódicamente. Si se resuelve el problema que ha impedido la recuperación, el objeto se recuperará en el siguiente reintento, o el objeto se puede recuperar manualmente, utilizando el mandato **rcrmqobj** .

Protección de los archivos de registro de IBM MQ

No toque los archivos de registro cuando un gestor de colas esté en ejecución, puede provocar que la recuperación no sea posible. Utilice autorización mqm o de superusuario para proteger los archivos de registro de modificaciones involuntarias.

No elimine manualmente los archivos de registro activos cuando un gestor de colas de IBM MQ esté en ejecución. Si un usuario suprime accidentalmente los archivos de registro que un gestor de colas necesita para reiniciarse, IBM MQ **no** emite ningún error y sigue procesando datos, *incluidos los mensajes persistentes*. El gestor de colas concluye normalmente, pero puede no reiniciarse. La recuperación de mensajes será entonces imposible.

Los usuarios que tienen la autorización para eliminar registros que esté utilizando un gestor de colas activo también tienen autorización para suprimir otros recursos importantes del gestor de colas (como archivos de colas, el catálogo de objetos y los archivos ejecutables de IBM MQ). Estos usuarios pueden, por tanto, dañar, quizá por falta de experiencia, un gestor de colas en ejecución o inactivo de una forma contra la cual IBM MQ no puede protegerse.

Tenga cuidado cuando otorgue autorizaciones mqm o de superusuario.

Volcado del contenido del registro mediante el mandato dmpmqlog

Cómo utilizar el mandato dmpmqlog para realizar un vuelco del contenido de las anotaciones del gestor de colas.

Utilice el mandato dmpmqlog para volcar el contenido de las anotaciones del gestor de colas. De forma predeterminada, se vuelcan todos los registros de anotaciones activos, es decir, que el mandato empieza a volcar desde la cabeza del registro (generalmente el principio del último punto de comprobación completado).

Normalmente, el registro sólo pueden volcarse cuando el gestor de colas no está ejecutándose. Como el gestor de colas efectúa un punto de comprobación durante el cierre, la parte activa del registro contiene normalmente un número reducido de registros de anotaciones. No obstante, puede utilizar el mandato dmpmqlog para realizar un volcado de más registros de anotaciones con una de las siguientes opciones que permiten cambiar la posición inicial del volcado:

- El vuelco inicial desde la *base* de las anotaciones. La base del registro es el primer registro de anotaciones del archivo de registro que contiene la cabeza del registro. La cantidad de datos adicionales volcados en este caso depende del lugar del archivo de registro dónde se encuentre la cabeza del registro. Si es cerca del principio del archivo de registro, se volcará una pequeña cantidad de datos adicionales. Si la cabeza está cerca del final del archivo de registro, se volcarán bastantes más datos.
- Especifique la posición inicial del volcado como un registro de anotaciones individual. Cada registro de anotaciones se identifica mediante un *número de secuencia de anotaciones* exclusivo. En el caso del registro circular, este registro de anotaciones inicial no puede ser anterior a la base del registro; esta limitación no se aplica al registros lineales. Es posible que deba volver a crear una instancia de los archivos de registro inactivos antes de ejecutar el mandato. Como posición inicial debe especificar un LSN válido, tomado de la salida anterior del mandato `dmpmqlog`.

  Por ejemplo, con las anotaciones cronológicas lineales puede especificar `nextlsn` de la última salida generada por `dmpmqlog`. El `nextlsn` aparece en `Log File Header` e indica el LSN del siguiente registro de anotaciones que se va a grabar. Utilícelo como posición inicial para formatear todos los registros de anotaciones que se hayan grabado desde la última vez que se efectuó un volcado del registro.
- **Sólo para anotaciones lineales**, puede indicar a `dmpmqlog` que inicie el formateo de los registros de anotaciones a partir de la extensión de archivo de anotaciones que desee. En ese caso, `dmpmqlog` espera encontrar este archivo de anotaciones, y los archivos posteriores, en el mismo directorio que los archivos de anotaciones activos. Esta opción no se aplica a las anotaciones circulares, en las `dmpmqlog` no puede acceder a los registros de anotaciones antes de acceder a la base de las anotaciones.

La salida del mandato `dmpmqlog` es la cabecera del archivo de anotaciones (`Log File Header`) y una serie de registros de anotaciones formateados. El gestor de colas utiliza varios registros de anotaciones para registrar los cambios efectuados en sus datos.

Parte de la información que se formatea se utiliza tan solo internamente. La siguiente lista incluye los registros de anotaciones más útiles:

Log File Header
Cada archivo de anotaciones tiene una sola cabecera de archivo de anotaciones, que es siempre el primer elemento formateado por el mandato `dmpmqlog`. Contiene los campos siguientes:

logactive	El número de extensiones de las anotaciones primarias.
loginactive	El número de extensiones de registro secundario.
logsize	El número de páginas de 4 KB por extensión.
baselsn	El primer LSN de la extensión de registro que contiene la cabeza del registro.
nextlsn	El LSN del siguiente registro de anotaciones que va a grabarse.
headlsn	El LSN del registro de anotaciones de la cabeza del registro.
tailsn	El LSN que identifica la posición de cola del registro.
hflag1	Indica si el registro es CIRCULAR (circular) o LOG RETAIN (lineal).
HeadExtentID	La extensión de registro que contiene la cabeza del registro.

Log Record Header
Cada registro de anotaciones del registro tiene una cabecera fija que contiene la siguiente información:

LSN	El número de secuencia de las anotaciones.
LogRecdType	El tipo de registro de anotaciones.

XTranid El identificador de transacción asociado a este registro de anotaciones (si lo hay).

Un *TranType* de MQI indica una transacción solo de IBM MQ. Un *TranType* XA está asociado a otros gestores de recursos. Las actualizaciones implicadas en la misma unidad de trabajo tienen el mismo *XTranid*.

QueueName La cola asociada a este registro de anotaciones (si lo hay).

Qid El identificador interno y exclusivo de la cola.

PrevLSN El LSN del registro de anotaciones anterior dentro de la misma transacción (si lo hay).

Start Queue Manager
Se anota que el gestor de colas se ha iniciado.

StartDate La fecha en que se inició el gestor de colas.

StartTime La hora en que se inició el gestor de colas.

Stop Queue Manager
Se anota que el gestor de colas se ha detenido.

StopDate La fecha en que se detuvo el gestor de colas.

StopTime La hora en que se detuvo el gestor de colas.

ForceFlag El tipo de cierre utilizado.

Start Checkpoint
Indica el inicio de un punto de comprobación del gestor de colas.

End Checkpoint
Indica el final de un punto de comprobación del gestor de colas.

ChkPtLSN El LSN del registro de anotaciones que se inició en este punto de comprobación.

Put Message
Anota que se ha transferido un mensaje persistente a una cola. Si el mensaje se transfirió bajo el punto de sincronización, la cabecera del registro de anotaciones contendrá un *XTranid* que no sea nulo. El resto del registro contiene:

MapIndex Un identificador del mensaje en la cola. Puede utilizarse para buscar la MQGET correspondiente utilizada para obtener el mensaje de la cola. En este caso, podrá encontrarse un registro de anotaciones *Get Message* posterior con el mismo QueueName y MapIndex. Al llegar a este punto, el identificador MapIndex podrá volver a utilizarse para una transferencia de mensaje subsiguiente a esta cola.

Datos Dentro del vuelco hex para este registro de anotaciones se encuentran varios datos internos, seguidos por una representación del descriptor de mensaje (MD resaltado) y del mensaje de datos propiamente dicho.

Put Part
Los mensajes persistentes que son demasiado grandes para un solo registro de anotaciones se anotan como varios registros de anotaciones *Put Part* seguidos de un solo registro *Put Message*. Si hay registros *Put Part*, el campo *PrevLSN* encadenará los registros *Put Part* y el último registro *Put Message*.

Datos Continúa los datos del mensaje donde terminó el registro de anotaciones anterior.

Get Message
Sólo se anotan las obtenciones de mensajes persistentes. Si el mensaje se transfirió bajo el punto de sincronización, la cabecera del registro de anotaciones contendrá un *XTranid* que no sea nulo. El resto del registro contiene:

MapIndex Identifica el mensaje que se recuperó de la cola. El registro de anotaciones *Put Message* más reciente que contiene el mismo *QueueName* y *MapIndex* identifica el mensaje que se ha recuperado.

QPriority La prioridad del mensaje recuperado de la cola.

Start Transaction
Indica el inicio de una nueva transacción. Un TranType de MQI indica una transacción sólo de IBM MQ. Un TranType XA indica una transacción que está asociada a otros gestores de recursos. Todas las actualizaciones efectuadas por esta transacción tendrán el mismo *XTranid*.

Prepare Transaction
Indica que el gestor de colas está preparado para confirmar las actualizaciones asociadas al *XTranid* especificado. Este registro de anotaciones se graba como parte de una confirmación en dos fases que implique a otros gestores de recursos.

Commit Transaction
Indica que el gestor de colas ha confirmado todas las actualizaciones efectuadas por una transacción.

Retrotraer transacción
Este registro de anotaciones indica la intención del gestor de colas de restituir una transacción.

End Transaction
Indica el final de una transacción restituida.

Transaction Table
Este registro se graba durante el punto de sincronización. Registra el estado de cada transacción que haya realizado actualizaciones persistentes. Se registra la siguiente información para cada transacción:

XTranid El identificador de la transacción.

FirstLSN El LSN del primer registro de anotaciones asociado a la transacción.

LastLSN El LSN del último registro de anotaciones asociado a la transacción.

Transaction Participants
Este registro de anotaciones lo graba el componente Gestor de transacciones de XA del gestor de colas. Registra los gestores de recursos externos que participan en las transacciones. Se registra lo siguiente para cada participante:

RMName El nombre del gestor de recursos.

RMID El identificador del gestor de recursos. Este identificador se anota también en los registros de anotaciones *Transaction Prepared* posteriores que registren transacciones globales en las que participe el gestor de recursos.

SwitchFile El archivo de carga conmutada de este gestor de recursos.

XAOpenString La serie de apertura de XA para este gestor de recursos.

XACloseString La serie de cierre de XA de este gestor de recursos.

Transaction Prepared
Este registro de anotaciones lo graba el componente Gestor de transacciones de XA del gestor de colas. Indica que la transacción global especificada se ha preparado correctamente. Se pedirá

a cada gestor de recursos participante que efectúe una confirmación. El *RMID* de cada gestor de recursos preparado se anota en el registro de anotaciones. Si el propio gestor de colas participa en la transacción, habrá una entrada *Participant Entry* con un valor de *RMID* de cero.

Transaction Forget
Este registro de anotaciones lo graba el componente Gestor de transacciones de XA del gestor de colas. Sigue el registro de anotaciones de *Transaction Prepared* cuando la decisión de confirmación se ha entregado a cada participante.

Purge Queue
Anota el hecho de que se han eliminado todos los mensajes de una cola, por ejemplo, mediante el mandato MQSC CLEAR QUEUE.

Queue Attributes
Anota la inicialización o modificación de los atributos de una cola.

Create Object
Anota la creación de un objeto de IBM MQ.

ObjName	El nombre del objeto que se ha creado.
UserId	El ID de usuario que ha llevado a cabo la creación.

Delete Object
Anota la supresión de un objeto de IBM MQ.

ObjName	El nombre del objeto que se ha suprimido.

Copia de seguridad y restauración de datos de gestor de colas de IBM MQ

Se pueden proteger los gestores de colas frente a posibles corrupciones producidas por fallos de hardware haciendo copias de seguridad de los gestores de colas y de sus datos, haciendo una copia de seguridad solo de la configuración del gestor de colas y usando un gestor de colas de copia de seguridad.

Acerca de esta tarea

PRECAUCIÓN: Debe tener mucho cuidado si mueve un gestor de colas a un sistema operativo distinto. Para obtener más información, consulte Mover un gestor de colas a un sistema operativo distinto.

Periódicamente, puede tomar medidas para proteger los gestores de colas contra posibles daños producidos por anomalías en el hardware. Existen tres formas de proteger un gestor de colas:

Copiar los datos del gestor de colas
Si el hardware falla, un gestor de colas puede verse forzado a detenerse. Si se pierden datos de registro del gestor de colas debido a la anomalía de hardware, el gestor de colas podría no reiniciarse. Si hace una copia de seguridad de los datos del gestor de colas, es posible que pueda recuperar algunos, o todos, los datos de gestor de colas perdidos.

En general, cuanto mayor sea la frecuencia a la que realiza la copia de seguridad de los datos, menos datos perderá en el caso de una anomalía en el hardware que provoque la pérdida de integridad en el registro de recuperación.

Para realizar una copia de los datos del gestor de colas, el gestor de colas no debe estar en ejecución.

Realice una copia de la configuración del gestor de colas solamente.
Si el hardware falla, un gestor de colas puede verse forzado a detenerse. Si se pierden los datos de registro cronológico y de configuración del gestor de colas por un fallo de hardware, el gestor de colas no podrá reiniciarse ni recuperarse a partir del registro cronológico. Si se hace una copia de seguridad de la configuración del gestor de colas, se puede volver a crear dicho gestor y todos sus objetos a partir de las definiciones guardadas.

Para realizar una copia de seguridad de la configuración del gestor de colas, éste debe estar en ejecución.

Utilizar un gestor de colas de copia de seguridad

Si el error de hardware es grave, un gestor de colas puede ser irrecuperable. En esta situación, si el gestor de colas irrecuperable tiene un gestor de colas de copia de seguridad dedicado, se puede activar el gestor de colas de copia de seguridad en lugar del gestor de colas irrecuperable. Si se actualiza regularmente, el registro cronológico del gestor de colas de copia de seguridad puede contener datos de registro que incluyan el último registro completo del gestor de colas irrecuperable.

Un gestor de colas de copia de seguridad se puede actualizar mientras se está ejecutando el gestor de colas existente.

Procedimiento

- Para realizar una copia de seguridad y restauración de datos del gestor de colas consulte:
  - "Hacer copia de seguridad de los datos de gestor de colas" en la página 707.
  - "Restauración de datos del gestor de colas" en la página 708.
- Para realizar una copia de seguridad y restauración de la configuración del gestor de colas, consulte:
  - "Copia de seguridad de la configuración del gestor de colas" en la página 709
  - "Restauración de la configuración del gestor de colas" en la página 710
- Para crear, actualizar e iniciar un gestor de colas de copia de seguridad, consulte "Utilizar un gestor de colas de copia de seguridad" en la página 710.

Hacer copia de seguridad de los datos de gestor de colas

Hacer una copia de seguridad de los datos de gestor de colas puede ayudarle a evitar la posible perdida de datos debida a errores de hardware.

Antes de empezar

Antes de empezar a hacer la copia de seguridad del gestor de colas, asegúrese de que el gestor de colas no esté ejecutando. Si intenta hacer una copia de seguridad de un gestor de colas en ejecución, puede que la copia de seguridad no sea coherente debido a posibles actualizaciones en curso en el momento de copiar los archivos. Si es posible, detenga el gestor de colas ejecutando el mandato **endmqm -w** (una conclusión de espera), solo si falla, utilice el mandato **endmqm -i** (una conclusión inmediata).

Acerca de esta tarea

Para realizar una copia de seguridad de los datos de un gestor de colas, realice estas tareas:

Procedimiento

1. Busque los directorios en los que el gestor de colas coloca los datos y los archivos de registro, utilizando la información de los archivos de configuración.

   Para obtener más información, consulte "Cambio de la información de configuración de IBM MQ en archivos .ini en Multiplatforms" en la página 95.

   Nota: Los nombres que aparecen en el directorio se transforman para asegurar que sean compatibles con la plataforma en la que se está utilizando IBM MQ. Para obtener más información sobre la transformación de nombres, consulte Comprender los nombres de archivo IBM MQ.
2. Haga copias de todos los directorios de archivos de datos y de registro del gestor de colas, incluidos todos los subdirectorios.

   Asegúrese de que no se deja ningún archivo, especialmente el archivo de control de registro, tal como se describe en "Cómo son los registros" en la página 673, y los archivos de configuración tal como se describen en "Archivos de inicialización y configuración" en la página 259. Algunos de los directorios pueden estar vacíos, pero todos son necesarios para restaurar la copia de seguridad posteriormente.

En el registro circular, haga copia de seguridad de los directorios de archivos de datos y de registro del gestor de colas al mismo tiempo para que pueda restaurar un conjunto coherente de datos y registros del gestor de colas.

En el registro lineal, haga copia de seguridad de los directorios de archivos de datos y de registro del gestor de colas al mismo tiempo. Es posible restaurar sólo los archivos de datos del gestor de colas si está disponible una secuencia completa de los archivos de registro correspondientes.

3. Mantenga la propiedad de los archivos.

   Para IBM MQ for UNIX y sistemas Linux, puede hacer esto con el mandato **tar**. Si tiene colas con más de 2 GB, no puede utilizar el mandato **tar**. Para obtener más información, consulte Habilitación de colas grandes.

   Nota: Cuando actualice a IBM WebSphere MQ 7.5 y posterior, asegúrese de realizar una copia de seguridad del archivo `qm.ini` y de las entradas del registro. La información del gestor de colas se almacena en el archivo `qm.ini` y se puede utilizar para revertir a una versión anterior de IBM MQ.

Tareas relacionadas

Detención de un gestor de colas

“Copia de seguridad de los archivos de configuración después de crear un gestor de colas” en la página 14

La información de configuración de IBM MQ se almacena en los archivos de configuración en AIX, Linux, and Windows. Después de crear un gestor de colas, haga una copia de seguridad de los archivos de configuración. A continuación, si crea otro gestor de colas que le causa algún problema, puede reinstalar las copias de seguridad cuando haya eliminado la causa del problema.

Restauración de datos del gestor de colas

Siga estos pasos para restaurar una copia de seguridad de los datos de un gestor de colas.

Antes de empezar

Antes de empezar la copia de seguridad, asegúrese de que el gestor de colas no está en ejecución.

Al restaurar una copia de seguridad de un gestor de colas en un clúster, consulte “Recuperación de un gestor de colas de clúster” en la página 388 y Agrupación en clúster: disponibilidad, varias instancias y recuperación tras desastre para obtener más información.

Nota: Cuando actualice a una versión posterior de IBM MQ, asegúrese de realizar una copia de seguridad del archivo **.ini** y de las entradas del registro. La información del gestor de colas se almacena en el archivo **.ini** y se puede utilizar para revertir a una versión anterior de IBM MQ.

Procedimiento

1. Localice los directorios en los que el gestor de colas coloca los datos y los archivos de registro, utilizando la información de los archivos de configuración.
2. Vacíe los directorios en los que va a colocar los datos de los que se ha hecho copia de seguridad.
3. Copie los datos y los archivos de registro del gestor de colas de los que se ha hecho copia de seguridad en los lugares correctos.

   Asegúrese de que tiene un archivo de control de registro, además de los archivos de registro.

   En el registro circular, haga copia de seguridad de los directorios de archivos de datos y de registro del gestor de colas al mismo tiempo para que pueda restaurar un conjunto coherente de datos y registros del gestor de colas.

   En el registro lineal, haga copia de seguridad de los directorios de archivos de datos y de registro del gestor de colas al mismo tiempo. Es posible restaurar sólo los archivos de datos del gestor de colas si está disponible una secuencia completa de los archivos de registro correspondientes.
4. Actualice los archivos de información de configuración.

Compruebe que los archivos de configuración de IBM MQ y del gestor de colas sean coherentes para que IBM MQ pueda buscar los datos restaurados en el lugar correcto.

5. Compruebe la estructura de directorios resultante para asegurarse de que tiene todos los directorios necesarios.

   Para obtener más información sobre los directorios y subdirectorios de IBM MQ, consulte Estructura de directorios en sistemas Windows y Contenido de directorio en sistemas AIX and Linux.

Resultados

Si tanto la copia de seguridad como la restauración de los datos se ha realizado correctamente, el gestor de colas debería iniciarse ahora.

Multi Copia de seguridad de la configuración del gestor de colas

Una copia de seguridad de la configuración del gestor de colas puede ayudar a reconstruir un gestor de colas a partir de su definiciones si se pierden los datos de registro cronológico y de configuración del gestor de colas por un fallo de hardware y el gestor de colas no puede reiniciarse ni recuperarse a partir del registro cronológico.

Acerca de esta tarea

ALW En AIX, Linux, and Windows, puede utilizar el mandato **dmpmqcfg** para volcar la configuración de un gestor de colas de IBM MQ.

IBM i En IBM i, puede utilizar el mandato de volcado de configuración de MQ (**DMPMQMCFG**) para volcar los objetos de configuración y autorizaciones de un gestor de colas.

Procedimiento

1. Asegúrese de que el gestor de colas esté ejecutando.
2. Dependiendo de la plataforma, utilice uno de los mandatos siguientes para hacer una copia de seguridad de la configuración del gestor de colas:

   - ALW En AIX, Linux, and Windows: Ejecute el mandato de volcado de configuración de MQ, **dmpmqcfg**, utilizando la opción de formato predeterminada (-f mqsc) MQSC y todos los atributos (-a), y utilice la redirección de la salida estándar para almacenar las definiciones en un archivo. Por ejemplo:

     ```
     dmpmqcfg -m MYQMGR -a > /mq/backups/MYQMGR.mqsc
     ```

   - IBM i En IBM i: Ejecute el mandato de volcado de configuración de MQ (**DMPMQMCFG**) utilizando la opción de formato predeterminada de OUTPUT(*MQSC) y EXPATTR(*ALL), y utilice TOFILE y TOMBR para almacenar las definiciones en un miembro de archivo físico. Por ejemplo:

     ```
     DMPMQMCFG MQMNAME(MYQMGR) OUTPUT(*MQSC) EXPATTR(*ALL) TOFILE(QMQMSAMP/QMQSC)
     TOMBR(MYQMGRDEF)
     ```

Tareas relacionadas

“Restauración de la configuración del gestor de colas” en la página 710
Para restaurar la configuración de un gestor de colas a partir de una copia de seguridad, primero asegúrese de que el gestor de colas esté ejecutando y luego ejecute el correspondiente mandato de su plataforma.

Referencia relacionada

dmpmqcfg (volcar configuración del gestor de colas)
Volcado de configuración de MQ (DMPMQMCFG)

Multi Restauración de la configuración del gestor de colas

Para restaurar la configuración de un gestor de colas a partir de una copia de seguridad, primero asegúrese de que el gestor de colas esté ejecutando y luego ejecute el correspondiente mandato de su plataforma.

Acerca de esta tarea

ALW En AIX, Linux, and Windows, puede utilizar el mandato **runmqsc** para restaurar la configuración de un gestor de colas de IBM MQ.

IBM i En IBM i, se puede usar el mandato **STRMQMMQSC** para restaurar los objetos de configuración y las autoridades de un gestor de colas.

Procedimiento

1. Asegúrese de que el gestor de colas esté ejecutando.

   Tenga en cuenta que, si el daño a los datos y los registros es irreparable por otros medios, puede que haya que volver a crear el gestor de colas.
2. Dependiendo de la plataforma, utilice uno de los mandatos siguientes para restaurar la configuración del gestor de colas:

   - ALW En AIX, Linux, and Windows, ejecute **runmqsc** en el gestor de colas, utilice la redirección de entrada estándar para restaurar las definiciones a partir de un archivo de script generado por el mandato de volcado de configuración de MQ (**dmpmqcfg**) (consulte "Copia de seguridad de la configuración del gestor de colas" en la página 709). Por ejemplo:

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

   - IBM i En IBM i: Ejecute **STRMQMMQSC** en el gestor de colas, y utilice los parámetros **SRCMBR** y **SRCFILE** para restaurar las definiciones del miembro de archivo físico generado por el mandato de volcado de configuración de MQ (**DMPMQMCFG**) (consulte "Copia de seguridad de la configuración del gestor de colas" en la página 709). Por ejemplo:

     ```
     STRMQMMQSC MQMNAME(MYQMGR) SRCFILE(QMQMSAMP/QMQSC) SRCMBR(MYQMGR)
     ```

Tareas relacionadas

"Copia de seguridad de la configuración del gestor de colas" en la página 709
Una copia de seguridad de la configuración del gestor de colas puede ayudar a reconstruir un gestor de colas a partir de su definiciones si se pierden los datos de registro cronológico y de configuración del gestor de colas por un fallo de hardware y el gestor de colas no puede reiniciarse ni recuperarse a partir del registro cronológico.

Referencia relacionada

dmpmqcfg (volcar configuración del gestor de colas)
runmqsc (ejecutar mandatos MQSC)
Volcado de configuración de MQ (DMPMQMCFG)
Mandatos de inicio de IBM MQ (STRMQMMQSC)

Utilizar un gestor de colas de copia de seguridad

Un gestor de colas existente puede tener un gestor de colas de copia de seguridad dedicado para fines de recuperación tras desastre.

Acerca de esta tarea

Un gestor de colas de copia de seguridad es una copia inactiva del gestor de colas existente. Si el gestor de colas existente no se puede recuperar debido a una anomalía grave del hardware, se puede poner en línea el gestor de colas de copia de seguridad para que sustituya al gestor de colas irrecuperable.

Los archivos de registro de gestor de colas existentes deben copiarse con cierta frecuencia en el gestor de colas de copia de seguridad para garantizar que éste resulte un método eficaz para la recuperación de errores. No es necesario detener el gestor de colas existente para que se copien los archivos de registro; sin embargo, sólo debe copiar un archivo de registro si el gestor de colas ha terminado de escribir en él; consulte "Actualización de un gestor de colas de copia de seguridad" en la página 712 para obtener información sobre cómo asegurarse de que un archivo de registro específico ya no se está grabando en él, para que se pueda copiar de forma segura.

Nota: Dado que el registro del gestor de colas existente se actualiza continuamente, siempre hay un grado de discrepancia entre el registro del gestor de colas existente y los datos de registro que se copian en el registro del gestor de colas de copia de seguridad. Las actualizaciones regulares del gestor de colas de copia de seguridad minimiza la discrepancia entre los dos registros.

Si desea poner en línea un gestor de colas de copia de seguridad, primero debe activarlo e iniciarlo. El requisito para activar un gestor de colas de copia de seguridad antes de iniciarlo es una medida preventiva para evitar que el gestor de colas de copia de seguridad se inicie accidentalmente. Una vez que haya activado el gestor de colas de copia de seguridad ya no podrá actualizarlo.

Importante: Una vez que el gestor de colas de copia de seguridad se ha convertido en el nuevo gestor de colas activo, por el motivo que sea, deja de haber un gestor de colas de copia de seguridad. Esto es efectivamente una forma de réplica asíncrona, por lo que lógicamente cabe esperar que el nuevo gestor de colas quede por detrás del antiguo gestor de colas activo durante un tiempo. Como tal, el antiguo gestor de colas activo ya no actúa como una copia de seguridad del nuevo gestor de colas activo.

Procedimiento

- Para obtener información sobre cómo utilizar un gestor de colas de copia de seguridad, consulte los temas siguientes:
  - "Creación de un gestor de colas de copia de seguridad" en la página 711
  - "Actualización de un gestor de colas de copia de seguridad" en la página 712
  - "Inicio de un gestor de colas de copia de seguridad" en la página 713

Conceptos relacionados

"Registro: Asegurarse de que no se han perdido mensajes" en la página 673
IBM MQ registra todos los cambios significativos en los datos persistentes controlados por el gestor de colas en un registro de recuperación.

Creación de un gestor de colas de copia de seguridad

Un gestor de colas de copia de seguridad se crea como una copia interactiva del gestor de colas existente.

Acerca de esta tarea

Importante: Sólo puede utilizar un gestor de colas de copia de seguridad con las anotaciones cronológicas lineales.

Un gestor de colas de copia de seguridad requiere lo siguiente:

- Tener los mismos atributos que el gestor de colas existente, por ejemplo el nombre del gestor de colas, el tipo de registro cronológico y el tamaño del archivo de registro.
- Ejecutarse en la misma plataforma que el gestor de colas existente.
- Encontrarse en un nivel de código igual, o superior, al del gestor de colas existente.

Procedimiento

1. Cree un gestor de colas de copia de seguridad para el gestor de colas existente con el mandato de control **crtmqm**.
2. Haga copias de todos los datos del gestor de colas y los directorios de archivos de registro existentes, incluidos todos los subdirectorios. como se describe en “Hacer copia de seguridad de los datos de gestor de colas” en la página 707.
3. Sobrescriba los directorios de archivos de registro y de datos del gestor de colas de copia de seguridad, incluidos todos los subdirectorios, con las copias realizadas en el gestor de colas existente.
4. Ejecute el mandato de control **strmqm** en el gestor de colas de copia de seguridad tal y como se muestra en el ejemplo siguiente:

   ```
   strmqm -r BackupQMName
   ```

   Este mandato marca el gestor de colas como un gestor de colas de copia de seguridad en IBM MQ y reproduce todas las extensiones de registro copiadas para sincronizar el gestor de colas de copia de seguridad con el gestor de colas existente.

Referencia relacionada

crtmqm (crear gestor de colas)

strmqm (iniciar gestor de colas)

Actualización de un gestor de colas de copia de seguridad

Para asegurar que un gestor de colas de copia de seguridad sigue siendo un método eficaz para la recuperación de errores, debe actualizarlo con cierta frecuencia.

Acerca de esta tarea

Las actualizaciones regulares minimizan la discrepancia entre el registro del gestor de colas de copia de seguridad y el registro del gestor de colas actual. No es necesario parar el gestor de colas para hacer una copia de seguridad del mismo.

Aviso: Si se copia un conjunto de registros cronológicos no correlativos en el directorio de registros del gestor de colas de copia de seguridad, solo se reproducirán los registros hasta el punto en que se encuentre el primer registro que falta.

Procedimiento

1. Emita el siguiente mandato de script (MQSC) en el gestor de colas del que se va a hacer una copia de seguridad:

   ```
   RESET QMGR TYPE(ADVANCELOG)
   ```

   Este mandato detiene cualquier grabación en las anotaciones actuales, y luego hace avanzar el registro cronológico del gestor de colas a la siguiente extensión de anotaciones. Esto garantiza que la copia de seguridad contenga toda la información registrada hasta el momento actual.
2. Obtenga el (nuevo) número de extensión de registro activo actual emitiendo el siguiente mandato de Script (MQSC) en el gestor de colas del que se va a hacer una copia de seguridad:

   ```
   DIS QMSTATUS CURRLOG
   ```

3. Copie los archivos de extensión de registro actualizados del directorio de registros del gestor de colas actual al directorio de registro del gestor de colas de copia de seguridad.

   Copie todas las extensiones de registro desde la última actualización hasta la extensión actual (sin incluirla) anotada en “2” en la página 712. Copie sólo los archivos de extensión de registro, los que empiezan por "S. ..".
4. Ejecute el mandato de control **strmqm** en el gestor de colas de copia de seguridad tal y como se muestra en el ejemplo siguiente:

```
strmqm -r BackupQMName
```

Este mandato reproduce todas las extensiones de registro copiadas y sincroniza el gestor de colas de copia de seguridad con el gestor de colas. Cuando la reproducción finalice, recibirá un mensaje que identifica todas las extensiones de registro necesarias para la recuperación de reinicio y todas las extensiones de registro necesarias para la recuperación desde soporte.

Referencia relacionada

RESET QMGR

DISPLAY QMSTATUS

strmqm (iniciar gestor de colas)

Inicio de un gestor de colas de copia de seguridad

Puede sustituir un gestor de colas irrecuperable con un gestor de colas de copia de seguridad.

Acerca de esta tarea

Al restaurar una copia de seguridad de un gestor de colas en un clúster, consulte "Recuperación de un gestor de colas de clúster" en la página 388 y Agrupación en clúster: disponibilidad, varias instancias y recuperación tras desastre para obtener más información.

Si un gestor de colas irrecuperable tiene un gestor de colas de copia de seguridad dedicado, se puede activar el gestor de colas de copia de seguridad en lugar del gestor de colas irrecuperable.

Cuando se sustituye un gestor de colas irrecuperable por un gestor de colas de copia de seguridad, se pueden perder algunos de los datos del gestor de colas irrecuperable. La cantidad de datos perdidos depende de la fecha de la última actualización del gestor de colas de copia de seguridad. Cuanto más reciente sea la última actualización, menor será la pérdida de datos del gestor de colas.

Nota: Aunque los archivos de datos y de registro del gestor de colas estén contenidos en directorios distintos, asegúrese de hacer una copia de seguridad y de restaurar los directorios al mismo tiempo. Si los archivos de datos y de registro del gestor de colas no tiene la misma antigüedad, el gestor de colas no está en un estado válido y probablemente no se iniciará. Incluso si se inicia, es muy probable que los datos estén dañados.

Procedimiento

1. Ejecute el mandato de control **strmqm** para activar el gestor de colas de copia de seguridad, tal y como se muestra en el ejemplo siguiente:

   ```
   strmqm -a BackupQMName
   ```

   Se activa el gestor de colas de copia de seguridad. Una vez activado, el gestor de colas de copia de seguridad ya no se puede actualizar.
2. Ejecute el mandato de control **strmqm** para iniciar el gestor de colas de copia de seguridad, tal y como se muestra en el ejemplo siguiente:

   ```
   strmqm BackupQMName
   ```

   IBM MQ considera esta acción una recuperación de reinicio y utiliza el registro del gestor de colas de copia de seguridad. Durante la última actualización del gestor de colas de copia de seguridad se habrá producido una reproducción, por lo que solo se retrotraerán las transacciones activas a partir del último punto de comprobación registrado.
3. Reinicie todos los canales.
4. Compruebe la estructura de directorios resultante para asegurarse de que tiene todos los directorios necesarios.

   Para obtener más información sobre los directorios y subdirectorios de IBM MQ, consulte Planificación del soporte de sistema de archivos.

5. Asegúrese de que tiene un archivo de control de registro, además de los archivos de registro. Compruebe también que los archivos de configuración de IBM MQ y del gestor de colas sean coherentes para que IBM MQ pueda buscar los datos restaurados en el lugar correcto.

Resultados

Si tanto la copia de seguridad como la restauración de los datos se han realizado correctamente, el gestor de colas se iniciará.

Tareas relacionadas

"Reinicio de canales detenidos" en la página 251

Cuando un canal pasa al estado STOPPED, es preciso que reinicie el canal manualmente.

Referencia relacionada

strmqm (iniciar gestor de colas)

Multi Cambios en la recuperación de errores de clúster en servidores en Multiplatforms

El gestor de colas vuelve a ejecutar las operaciones que han causado problemas, hasta que se resuelven los problemas. Si, después de cinco días, los problemas no se resuelven, el gestor de colas concluye para impedir que la memoria caché quede anticuada.

El gestor de colas vuelve a ejecutar las operaciones que han causado problemas, hasta que se resuelven los problemas. Si, después de cinco días, los problemas no se resuelven, el gestor de colas concluye para impedir que la memoria caché quede anticuada. A medida que la memoria caché queda anticuada, aumenta el número de problemas.

Cada aspecto de la gestión de clústeres se maneja para un gestor de colas mediante el proceso del gestor de repositorios local, `amqrrmfa`. El proceso se ejecuta en todos los gestores de colas, aunque no haya ninguna definición de clúster.

IBM MQ, en lugar de detener el gestor de repositorios y seguir sin él, el gestor de repositorios vuelve a ejecutar las operaciones fallidas. Si el gestor de colas detecta un problema con el gestor de repositorios, sigue uno de los dos procedimientos siguientes.

1. Si el error no compromete el funcionamiento del gestor de colas, el gestor de colas escribe un mensaje en el registro de errores. Vuelve a ejecutar la operación fallida cada 10 minutos hasta que se realiza correctamente. De forma predeterminada, necesita cinco días para tratar el error; de lo contrario, el gestor de colas escribe un mensaje en el registro de errores y concluye. Puede posponer la conclusión cinco días.
2. Si el error compromete el funcionamiento del gestor de colas, el gestor de colas escribe un mensaje en el registro de errores y concluye inmediatamente.

Un error que compromete el funcionamiento del gestor de colas es un error que el gestor de colas no ha sido capaz de diagnosticar o un error que puede tener consecuencias imprevisibles. Este tipo de error suele dar como resultado que el gestor de colas escriba un archivo FFST. Los errores que comprometen la operación del gestor de colas pueden ser causados por un error en IBM MQ o por un administrador o un programa que hagan algo inesperado como, por ejemplo, finalizar un proceso de IBM MQ .

El punto del cambio en el comportamiento de recuperación de errores es limitar el tiempo que el gestor de colas continúa ejecutándose con un número creciente de definiciones de clúster inconsistentes. A medida que crece el número de inconsistencias en definiciones de clústeres, crece también la posibilidad de un comportamiento anómalo de las aplicaciones.

La posibilidad predeterminada de concluir el gestor de colas después de cinco días es un compromiso entre limitar el número de inconsistencias y mantener el gestor de colas disponible hasta que se detecten y se resuelvan los problemas.

Puede ampliar el tiempo antes de que el gestor de colas concluya de forma indefinida, mientras arregla el problema o espera que concluya un gestor de colas planificado. La permanencia de cinco días mantiene el gestor de colas en ejecución a lo largo de toda una semana y le ofrece tiempo para reaccionar a los problemas o prolongar el tiempo antes de reiniciar el gestor de colas.

Acciones correctivas

Para tratar los problemas de recuperación de errores de clústeres dispone de varias opciones. La primera opción es supervisar y solucionar el problema y la segunda es supervisar y posponer la solución del problema.

1. Supervise el registro de errores del gestor de colas para ver si se encuentran los mensajes de error AMQ9448 y AMQ5008 y solucione el problema.

   AMQ9448 indica que el gestor de repositorios ha devuelto un error después de ejecutar un mandato. Este error marca el inicio de volver a intentar el mandato cada 10 minutos y finalmente detener el gestor de colas después de cinco días, hasta que pospone la conclusión.
   AMQ5008 indica que el gestor de colas se ha detenido porque falta un proceso IBM MQ. AMQ5008 se debe a que el gestor de repositorios se detiene después de cinco días. Si el gestor de repositorios se detiene, se detiene el gestor de colas.
2. Supervise el registro de errores del gestor de colas para ver si se encuentra el mensaje de error AMQ9448 y posponga la resolución del problema.

   Si inhabilita obtener mensajes de `SYSTEM.CLUSTER.COMMAND.QUEUE`, el gestor de repositorios deja de intentar ejecutar mandatos y continúa de forma indefinida sin procesar ningún trabajo. No obstante, los manejadores que el gestor de repositorios retiene en colas se liberan. Dado que el gestor de repositorios no se detiene, el gestor de colas no se detiene después de cinco días.
   Ejecutar un mandato MQSC para inhabilitar la obtención de mensajes de `SYSTEM.CLUSTER.COMMAND.QUEUE`:
   ```
   ALTER QLOCAL(SYSTEM.CLUSTER.COMMAND.QUEUE) GET(DISABLED)
   ```
   Para volver a recibir mensajes de `SYSTEM.CLUSTER.COMMAND.QUEUE` ejecute un mandato MQSC:
   ```
   ALTER QLOCAL(SYSTEM.CLUSTER.COMMAND.QUEUE) GET(ENABLED)
   ```

Consideración especial

Detener `amqrrmfa` en la IBM MQ hace que el gestor de colas se detenga, porque se considera una anomalía del gestor de colas. No debe detener el proceso `amqrrmfa` a menos que establezca el parámetro de ajuste del gestor de colas, `TolerateRepositoryFailure`.

Ejemplo

```
TuningParameters:
        TolerateRepositoryFailure=TRUE
```

Figura 86. Establezca `TolerateRepositoryFailure` en TRUE en qm.ini

Conceptos relacionados

“Archivos de configuración de gestores de colas, qm.ini” en la página 109
Un archivo de configuración del gestor de colas, `qm.ini`, contiene información relevante para un gestor de colas específico. Atributos que se pueden utilizar para modificar la configuración de un gestor de colas individual y sustituir lo valores de IBM MQ.

Configuración de recursos JMS y Jakarta Messaging

Una de las formas en las que una aplicación JMS o Jakarta Messaging puede crear y configurar los recursos que necesita para conectarse a IBM MQ y acceder a los destinos para enviar o recibir mensajes es utilizando la interfaz JNDI (Java Naming and Directory Interface) para recuperar objetos administrados de una ubicación dentro del servicio de nombres y directorios denominado espacio de nombres JNDI. Para que una aplicación JMS pueda recuperar objetos administrados de un espacio de nombres JNDI, primero debe crear los objetos administrados.

Acerca de esta tarea

JM 3.0 A partir de IBM MQ 9.3.0, Jakarta Messaging 3.0 está soportado para desarrollar nuevas aplicaciones. IBM MQ 9.3.0 y posteriores siguen dando soporte a JMS 2.0 para las aplicaciones existentes. No está soportado utilizar tanto la API de Jakarta Messaging 3.0 como la API de JMS 2.0 en la misma aplicación. Para obtener más información, consulte Utilización de clases de IBM MQ para JMS/Jakarta Messaging.

Puede crear y configurar objetos administrados en IBM MQ utilizando cualquiera de las dos siguientes herramientas:

Herramientas de administración de IBM MQ JMS y Jakarta Messaging
La herramienta de administración de IBM MQ JMS , **JMSAdmin**y la herramienta de administración de Jakarta Messaging , **JMS30Admin**, son herramientas de línea de mandatos que puede utilizar para crear y configurar objetos de IBM MQ JMS y Jakarta Messaging que se almacenan en LDAP, en un sistema de archivos local u otras ubicaciones. Las herramientas de administración de JMS y Jakarta Messaging utilizan una sintaxis similar a **runmqsc**y también dan soporte a los scripts.

Las herramientas de administración utilizan un archivo de configuración para establecer los valores de determinadas propiedades. Se suministra un archivo de configuración de ejemplo, que puede editar para adaptarlo a su sistema antes de empezar a utilizar la herramienta para configurar recursos de JMS. Para obtener más información sobre el archivo de configuración, consulte “Configuración de las herramientas JMSAdmin y JMS30Admin” en la página 723.

JMS 2.0 **IBM MQ Explorer**
Para JMS 2.0, puede utilizar IBM MQ Explorer para crear y administrar definiciones de objeto de JMS 2.0 que se almacenan en LDAP, en un sistema de archivos local u otras ubicaciones.

JM 3.0 Para Jakarta Messaging 3.0, no puede administrar JNDI utilizando IBM MQ Explorer. La administración JNDI está soportada por la variante Jakarta Messaging 3.0 de **JMSAdmin**, que es **JMS30Admin**.

Las aplicaciones IBM MQ JMS que se despliegan en WebSphere Application Server necesitan acceder a objetos de JMS desde el repositorio JNDI del servidor de aplicaciones. Por lo tanto, si utiliza la mensajería JMS entre WebSphere Application Server e IBM MQ, debe crear objetos en WebSphere Application Server que correspondan a los objetos que cree en IBM MQ.

JM 3.0 Aunque IBM MQ 9.3 y posteriores dan soporte a Jakarta Messaging 3.0, WebSphere Application Server no tiene actualmente un soporte equivalente. Por lo tanto, en WebSphere Application Server, configure los recursos de Java Message Service 2.0 .

IBM MQ Explorer y la herramienta de administración de IBM MQ JMS no se pueden utilizar para administrar los objetos de IBM MQ JMS que se almacenan en WebSphere Application Server. En su lugar, puede crear y configurar objetos administrados en WebSphere Application Server utilizando cualquiera de las dos siguientes herramientas:

Consola administrativa de WebSphere Application Server
La consola de administración de WebSphere Application Server es una herramienta basada en web que puede utilizar para gestionar los objetos de IBM MQ JMS en WebSphere Application Server.

Cliente de scripts de WebSphere Application Server
El cliente de scripts wsadmin de WebSphere Application Server proporciona mandatos especializados para administrar los objetos de IBM MQ JMS en WebSphere Application Server.

Si desea utilizar una aplicación JMS para acceder a los recursos de un gestor de colas IBM MQ desde WebSphere Application Server, debe utilizar el proveedor de mensajería IBM MQ en WebSphere Application Server, que contiene una versión de IBM MQ classes for JMS. El adaptador de recursos de IBM MQ que se suministra con WebSphere Application Server se utiliza en todas las aplicaciones que realizan mensajería JMS con el proveedor de mensajería IBM MQ. El adaptador de recursos de IBM MQ normalmente se actualiza automáticamente cuando se aplican fixpacks de WebSphere Application Server, pero si ha actualizado manualmente con anterioridad el adaptador de recursos, debe actualizar manualmente la configuración para asegurar que el mantenimiento se aplique correctamente.

Conceptos relacionados

Creación y configuración de fábricas de conexiones y destinos en una aplicación de IBM MQ Classes for JMS

Referencia relacionada

runmqsc (ejecutar mandatos MQSC)

Configurar fábricas de conexiones y destinos en un espacio de nombres JNDI

Las aplicaciones JMS y Jakarta Messaging acceden a objetos administrados en el servicio de nombres y directorios a través de JNDI (Java Naming and Directory Interface). Los objetos administrados JMS o Jakarta Messaging se almacenan en una ubicación dentro del servicio de nombres y directorios al que se hace referencia como espacio de nombres JNDI. Una aplicación JMS o Jakarta Messaging puede buscar los objetos administrados para conectarse a IBM MQ y acceder a los destinos para enviar o recibir mensajes.

Acerca de esta tarea

Las aplicaciones JMS o Jakarta Messaging buscan los nombres de los objetos JMS o Jakarta Messaging en el servicio de nombres y directorios utilizando contextos:

Contexto inicial
El contexto inicial define la raíz del espacio de nombres JNDI. Para cada ubicación del servicio de nombres y directorios, debe especificar un contexto inicial para proporcionar un punto de partida desde el que una aplicación JMS o Jakarta Messaging puede resolver los nombres de los objetos administrados en esa ubicación del servicio de nombres y directorios.

Subcontextos
Un contexto puede tener uno o más subcontextos. Un subcontexto es una subdivisión de un espacio de nombres JNDI y puede contener objetos administrados como fábricas de conexiones y destinos, así como otros subcontextos. Un subcontexto no es un objeto por sí mismo; es simplemente una extensión del convenio de denominación para los objetos en el subcontexto.

Para que una aplicación IBM MQ classes for JMS o IBM MQ classes for Jakarta Messaging pueda recuperar objetos administrados de un espacio de nombres JNDI, primero debe crear los objetos administrados. Puede crear y configurar los siguientes tipos de objeto JMS o Jakarta Messaging :

Fábrica de conexiones
Un objeto de fábrica de conexiones JMS o Jakarta Messaging define un conjunto de propiedades de configuración estándar para conexiones. Una aplicación JMS o Jakarta Messaging utiliza una fábrica de conexiones para crear una conexión con IBM MQ. Puede crear una fábrica de conexiones que sea específica de uno de los dos dominios de mensajería, el dominio de mensajería punto a punto y el dominio de mensajería de publicación/suscripción.

A partir de JMS 1.1, también puede crear fábricas de conexiones independientes del dominio que se pueden utilizar tanto para la mensajería punto a punto como para la mensajería de publicación/suscripción. Para obtener más información, consulte El modelo JMS y Jakarta Messaging.

Destino
Un destino JMS o Jakarta Messaging es un objeto que representa el destino de los mensajes que produce el cliente y el origen de los mensajes que consume una aplicación JMS . La aplicación JMSo Jakarta Messaging puede utilizar un único objeto de destino para colocar mensajes y obtener mensajes, o la aplicación puede utilizar objetos de destino independientes. Hay dos tipos de objeto de destino:

- Destino de cola JMS o Jakarta Messaging utilizado en la mensajería punto a punto
- Destino de tema JMS o Jakarta Messaging utilizado en la mensajería de publicación/suscripción

JMS 2.0 Para JMS 2.0, puede crear contextos y objetos administrados utilizando IBM MQ Explorer o la herramienta de administración de IBM MQ JMS **JMSAdmin**.

Nota: JM 3.0 Para Jakarta Messaging 3.0, no puede administrar JNDI utilizando IBM MQ Explorer. La administración JNDI está soportada por la variante Jakarta Messaging 3.0 de **JMSAdmin**, que es **JMS30Admin**.

El diagrama siguiente muestra un ejemplo de objetos JMS o Jakarta Messaging creados en un espacio de nombres JNDI de IBM MQ .

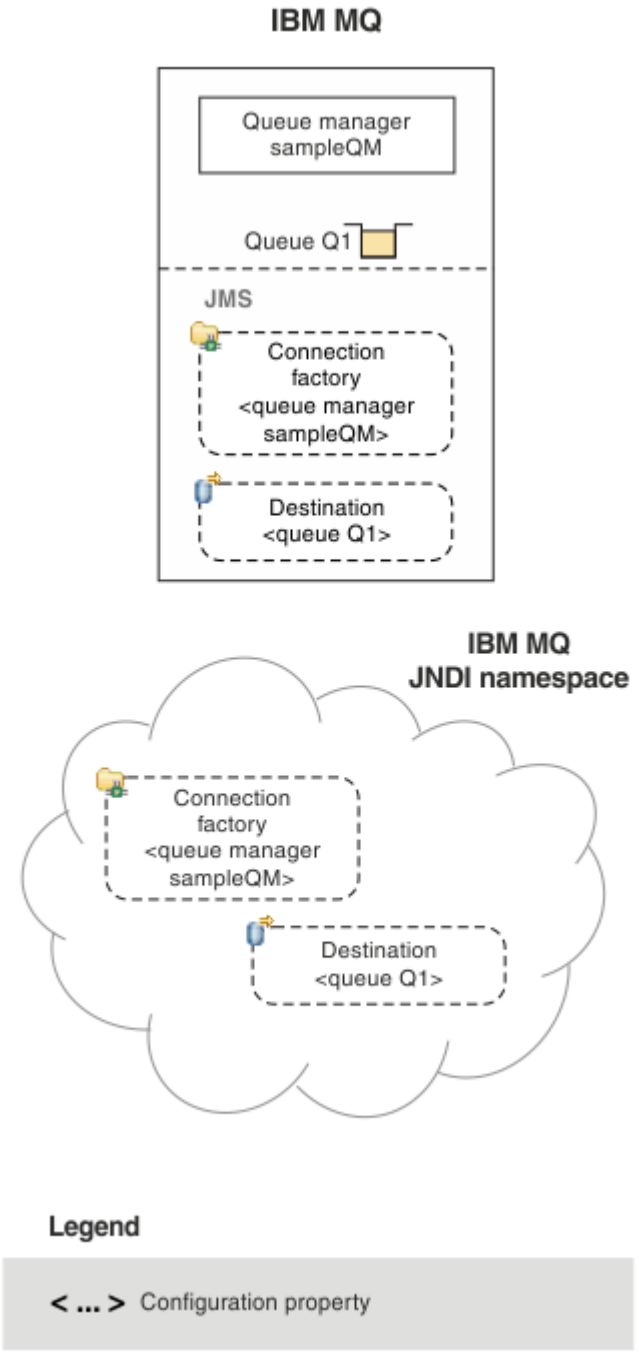


Figura 87. Objetos JMS o Jakarta Messaging creados en IBM MQ

Si utiliza la mensajería JMS entre WebSphere Application Server e IBM MQ, debe crear objetos correspondientes en WebSphere Application Server para utilizarlos para comunicarse con IBM MQ. Cuando se crea uno de estos objetos en WebSphere Application Server, se almacena en el espacio de nombres JNDI de WebSphere Application Server tal como se muestra en el siguiente diagrama.

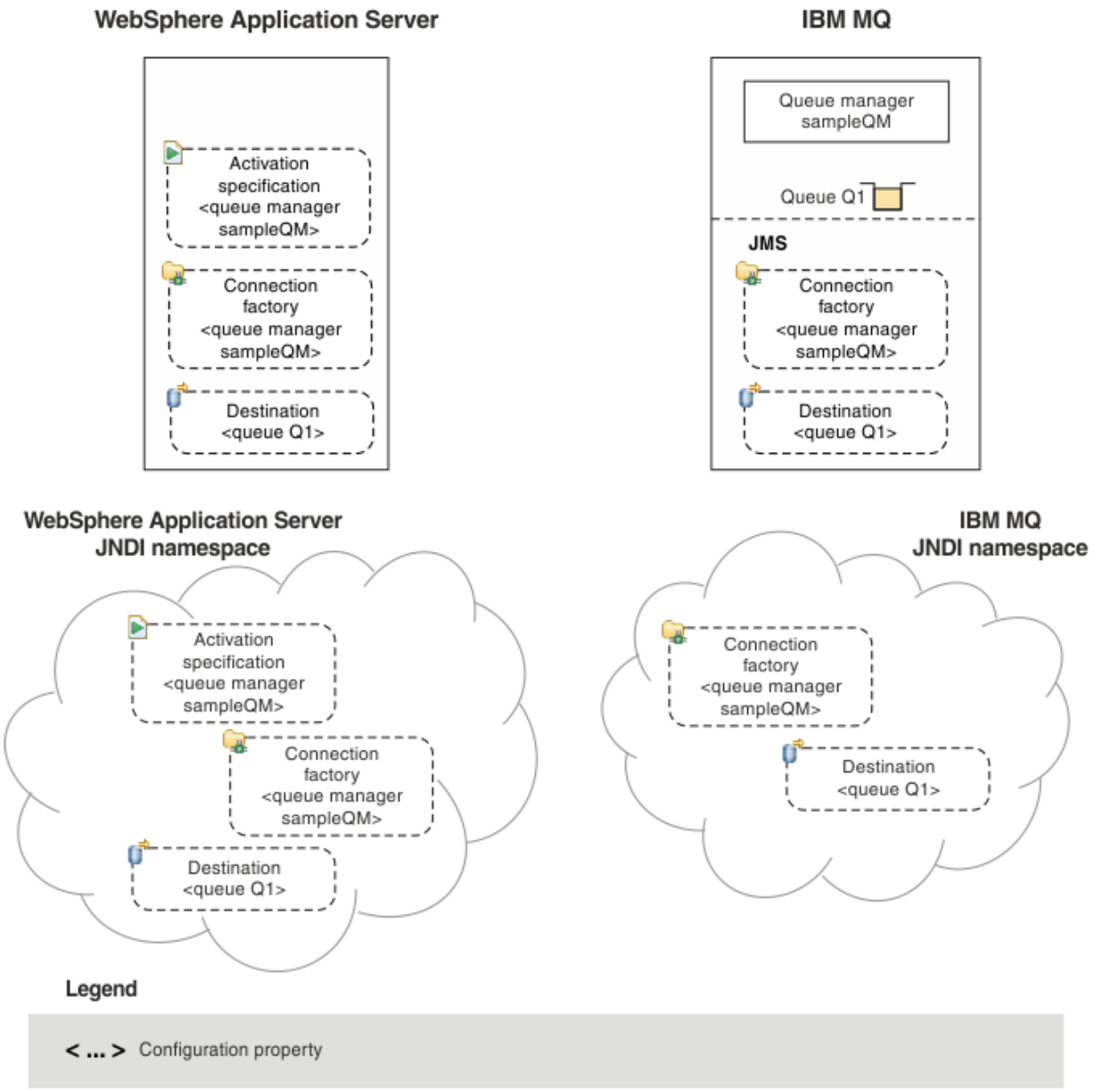


Figura 88. Objetos creados en WebSphere Application Server y los objetos correspondientes en IBM MQ

JMS 3.0 Aunque IBM MQ 9.3 y posteriores dan soporte a Jakarta Messaging 3.0, WebSphere Application Server no tiene actualmente un soporte equivalente. Por lo tanto, en WebSphere Application Server, configure los recursos de Java Message Service 2.0 .

Si su aplicación utiliza un bean controlado por mensaje (MDB), la fábrica de conexiones se utiliza sólo para mensajes de salida y los mensajes de entrada se reciben mediante una especificación de activación. Las especificaciones de activación forman parte del estándar Java EE Connector Architecture 1.5 (JCA 1.5). JCA 1.5 proporciona una forma estándar de integrar proveedores de JMS, como IBM MQ, con servidores de aplicaciones Java EE, como WebSphere Application Server. Una especificación de activación JMS puede estar asociada a uno o más beans controlados por mensaje (MDB) y proporciona la configuración necesaria para que estos MDB escuchen los mensajes que llegan a un destino.

Puede utilizar la consola administrativa de WebSphere Application Server o los mandatos de script wsadmin para crear y configurar los recursos de JMS que necesite.

Procedimiento

- JMS 2.0
  Para configurar objetos de JMS para IBM MQ utilizando IBM MQ Explorer, consulte "Configurar objetos JMS 2.0 utilizando IBM MQ Explorer" en la página 720.
- JMS 2.0
  Para configurar objetos de JMS para IBM MQ utilizando la herramienta de administración de IBM MQ JMS , **JMSAdmin**, consulte "Configuración de objetos JMS y Jakarta Messaging utilizando las herramientas de administración" en la página 721.
- JM 3.0
  Para configurar objetos de Jakarta Messaging para IBM MQ utilizando la herramienta de administración de IBM MQ Jakarta Messaging , **JMS30Admin**, consulte "Configuración de objetos JMS y Jakarta Messaging utilizando las herramientas de administración" en la página 721.
- JMS 2.0
  Para configurar objetos de JMS para WebSphere Application Server, consulte "Configurar recursos de JMS 2.0 en WebSphere Application Server" en la página 732.

Resultados

Una aplicación IBM MQ classes for JMS o IBM MQ classes for Jakarta Messaging puede recuperar los objetos administrados del espacio de nombres JNDI y, si es necesario, establecer o cambiar una o más de sus propiedades utilizando las extensiones IBM JMS o IBM MQ JMS .

Tareas relacionadas

Utilización de JNDI para recuperar objetos administrados en una aplicación JMS

Creación y configuración de fábricas de conexiones y destinos en una aplicación de IBM MQ classes for JMS

JMS 2.0 Configurar objetos JMS 2.0 utilizando IBM MQ Explorer

Utilice la interfaz gráfica de usuario IBM MQ Explorer para crear objetos JMS a partir de objetos de IBM MQ y objetos de IBM MQ a partir de objetos JMS, así como para administrar y supervisar otros objetos de IBM MQ.

Acerca de esta tarea

JMS 2.0 IBM MQ Explorer es la interfaz gráfica de usuario en la que puede administrar y supervisar objetos IBM MQ, independientemente de si están alojados en el sistema local o en un sistema remoto. IBM MQ Explorer se ejecuta en Windows y Linux for x86-64. Se puede conectar de forma remota a gestores de colas que se ejecuten en cualquier plataforma soportada incluyendo z/OS, lo que permite visualizar, explorar y modificar toda la estructura de mensajería desde la consola.

Nota: JM 3.0 Para Jakarta Messaging 3.0, no puede administrar JNDI utilizando IBM MQ Explorer. La administración JNDI está soportada por la variante Jakarta Messaging 3.0 de **JMSAdmin**, que es **JMS30Admin**.

En IBM MQ Explorer, todas las fábricas de conexiones se almacenan en carpetas Fábricas de conexiones en el contexto y subcontextos adecuados.

Puede realizar los siguientes tipos de tarea con IBM MQ Explorer, ya sea contextualmente a partir de un objeto existente en IBM MQ Explorer, o desde un asistente crear nuevo objeto:

- Crear una fábrica de conexiones JMS a partir de cualquiera de los siguientes objetos de IBM MQ:
  - Un gestor de colas IBM MQ, ya sea en el sistema local o en un sistema remoto.
  - Un canal IBM MQ.

  - Un escucha IBM MQ.
- Añadir un gestor de colas IBM MQ a IBM MQ Explorer utilizando una fábrica de conexiones JMS.
- Crear una cola JMS a partir de una cola IBM MQ.
- Crear una cola IBM MQ a partir de una cola JMS.
- Crear un tema JMS a partir de un tema IBM MQ, que puede ser un objeto de IBM MQ o un tema dinámico.
- Crear un tema IBM MQ a partir de un tema JMS.

Procedimiento

- Inicie IBM MQ Explorer, si aún no está en ejecución.

  Si IBM MQ Explorer está en ejecución y muestra la página de bienvenida, cierre la página de bienvenida para iniciar la administración de objetos IBM MQ.
- Si todavía no lo ha hecho, cree un contexto inicial que defina la raíz del espacio de nombres JNDI en el que se almacenan los objetos JMS en el servicio de nombres y directorio.

  Cuando haya añadido el contexto inicial a IBM MQ Explorer, puede crear objetos de fábrica de conexiones, objetos de destino y subcontextos en el espacio de nombres JNDI.

  El contexto inicial se muestra en la vista de Navegador en la carpeta Objetos administrados de JMS. Tenga en cuenta que aunque se muestra el contenido completo del espacio de nombres JNDI, en IBM MQ Explorer puede editar solamente los objetos de IBM MQ classes for JMS que están almacenados ahí. Para obtener más información, consulte Añadir un contexto inicial.
- Cree y configure los subcontextos y los objetos administrados de JMS que necesite.

  Para obtener más información, consulte Creación y configuración de objetos administrados de JMS.
- Configure IBM MQ.

  Para obtener más información, consulte Configurar IBM MQ utilizando IBM MQ Explorer .

Conceptos relacionados

Introducción a IBM MQ Explorer

Creación y configuración de fábricas de conexiones y destinos en una aplicación de IBM MQ classes for JMS

Configuración de objetos JMS y Jakarta Messaging utilizando las herramientas de administración

IBM MQ proporciona herramientas de administración que puede utilizar para definir las propiedades de ocho tipos de objeto IBM MQ classes for JMS o IBM MQ classes for Jakarta Messaging y para almacenarlas en un espacio de nombres JNDI. Las aplicaciones pueden luego utilizar JNDI para recuperar estos objetos administrados del espacio de nombres.

Acerca de esta tarea

JMS 2.0 Para JMS 2.0, la administración JNDI está soportada por la herramienta **JMSAdmin** .

JM 3.0 Para Jakarta Messaging 3.0, la administración JNDI está soportada por la variante Jakarta Messaging 3.0 de **JMSAdmin**, que es **JMS30Admin**.

La siguiente tabla muestra los ocho tipos de objetos administrados que puede crear, configurar y manipular utilizando verbos. La columna Palabra clave muestra las series que puede sustituir por *TYPE* en los mandatos que se muestran en Tabla 37 en la página 722.

Tabla 37. Los tipos de objeto JMS y Jakarta Messaging que maneja la herramienta de administración		
Tipo de objeto	**Palabra clave**	**Descripción**
MQConnectionFactory	CF	La implementación de IBM MQ de la interfaz ConnectionFactory de JMS. Representa un objeto de fábrica para crear conexiones en los dominios punto a punto y de publicación/suscripción.
MQQueueConnectionFactory	QCF	La implementación de IBM MQ de la interfaz QueueConnectionFactory de JMS. Representa un objeto de fábrica para crear conexiones en el dominio punto a punto.
MQTopicConnectionFactory	TCF	La implementación de IBM MQ de la interfaz TopicConnectionFactory de JMS. Representa un objeto de fábrica para crear conexiones en el dominio de publicación/suscripción.
MQQueue	Q	La implementación de IBM MQ de la interfaz Queue de JMS. Representa un destino para los mensajes en el dominio punto a punto.
MQTopic	T	La implementación de IBM MQ de la interfaz Topic de JMS. Representa un destino para los mensajes en el dominio de publicación/suscripción.
MQXAConnectionFactory "1" en la página 722	XACF	La implementación de IBM MQ de la interfaz XAConnectionFactory de JMS. Representa un objeto de fábrica para crear conexiones en los dominios punto a punto y de publicación/suscripción, y en el que las conexiones utilizan las versiones XA de las clases JMS.
MQXAQueueConnectionFábrica "1" en la página 722	XAQCF	La implementación de IBM MQ de la interfaz XAQueueConnectionFactory de JMS. Representa un objeto de fábrica para crear conexiones en el dominio punto a punto que utilizan las versiones XA de las clases JMS.
MQXATopicConnectionFábrica "1" en la página 722	XATCF	La implementación de IBM MQ de la interfaz XATopicConnectionFactory de JMS. Representa un objeto de fábrica para crear conexiones en el dominio de publicación/suscripción que utilizan las versiones XA de las clases JMS.

Nota:

1. Estas clases están destinadas a los proveedores de servidores de aplicaciones. Es poco probable que sean directamente útiles a los programadores de aplicaciones.

Para obtener más información sobre cómo configurar estos objetos, consulte "Configurar objetos JMS" en la página 731.

Los tipos y valores de propiedad que necesita para utilizar esta herramienta aparecen listados en Propiedades de objetos de IBM MQ classes for JMS.

También puede utilizar la herramienta para manipular subcontextos de espacio de nombres de directorio en JNDI como se describe en “Configurar subcontextos” en la página 727.

JMS 2.0 Para JMS 2.0 y anteriores, también puede crear y configurar objetos administrados de IBM MQ classes for JMS con IBM MQ Explorer.

JM 3.0 Para Jakarta Messaging 3.0, no puede administrar JNDI utilizando IBM MQ Explorer. La administración JNDI está soportada por la variante Jakarta Messaging 3.0 de **JMSAdmin**, que es **JMS30Admin**.

Conceptos relacionados

Creación y configuración de fábricas de conexiones y destinos en una aplicación de IBM MQ classes for JMS

Utilización de JNDI para recuperar objetos administrados en una aplicación JMS

Configuración de las herramientas JMSAdmin y JMS30Admin

Las herramientas de administración de IBM MQ JMS y Jakarta Messaging utilizan un archivo de configuración para establecer los valores de determinadas propiedades. En cada caso, se proporciona un archivo de configuración de ejemplo que puede editar para que se ajuste a su sistema.

Acerca de esta tarea

JM 3.0 IBM MQ 9.3.0 ha introducido soporte para Jakarta Messaging 3.0. JMS 2.0 sigue estando totalmente soportado.

El archivo de configuración es un archivo de texto sin formato que consta de un conjunto de pares de clave-valor, separados por el signo igual (=). Configure la herramienta de administración estableciendo valores para las tres propiedades definidas en el archivo de configuración. El siguiente ejemplo muestra estas tres propiedades:

```
#Set the service provider
INITIAL_CONTEXT_FACTORY=com.sun.jndi.ldap.LdapCtxFactory
#Set the initial context
PROVIDER_URL=ldap://polaris/o=ibm_us,c=us
#Set the authentication type
SECURITY_AUTHENTICATION=none
```

En este ejemplo, un signo de almohadilla (#) en la primera columna de la línea indica un comentario, o una línea que no se utiliza.

Con IBM MQ se suministra un archivo de configuración de ejemplo, que se utiliza como el archivo de configuración predeterminado. El archivo de ejemplo se denomina `JMSAdmin.config` (para JMS 2.0) o `JMS30Admin.config` (para Jakarta Messaging 3.0). Este archivo se encuentra en el directorio *MQ_JAVA_INSTALL_PATH*`/bin` . Puede editar el archivo de ejemplo para definir los valores necesarios para el sistema o crear su propio archivo de configuración.

Al iniciar la herramienta de administración, puede especificar el archivo de configuración que desea utilizar con el parámetro de línea de mandatos `-cfg`, tal como se describe en “Inicio de las herramientas JMSAdmin y JMS30Admin” en la página 725. Si no especifica un nombre de archivo de configuración al invocar la herramienta, la herramienta intenta cargar el archivo de configuración predeterminado (`JMSAdmin.config` o `JMS30Admin.config`). Busca este archivo primero en el directorio actual y, a continuación, en el directorio *MQ_JAVA_INSTALL_PATH*`/bin` , donde *MQ_JAVA_INSTALL_PATH* es la vía de acceso a la instalación de IBM MQ classes for JMS o IBM MQ classes for Jakarta Messaging .

Los nombres de los objetos JMS o Jakarta Messaging que se almacenan en un entorno LDAP deben cumplir con los convenios de denominación LDAP. Una de estas convenciones es que los nombres de objeto y de contexto deben incluir un prefijo, como `cn=` (nombre común) u `ou=` (unidad organizativa). La

herramienta de administración simplifica el uso de los proveedores de servicio LDAP al permitir hacer referencia a nombres de contexto y de objeto sin un prefijo. Si no se proporciona un prefijo, la herramienta añade automáticamente un prefijo predeterminado al nombre especificado. Para LDAP, este prefijo es cn=. Si es necesario, puede cambiar el prefijo predeterminado estableciendo la propiedad **NAME_PREFIX** en el archivo de configuración.

Nota: Es posible que tenga que configurar el servidor LDAP para almacenar objetos Java. Para obtener más información, consulte la documentación del servidor LDAP.

Procedimiento

1. Defina el proveedor de servicios utilizado por la herramienta configurando la propiedad **INITIAL_CONTEXT_FACTORY**.

   Los valores soportados para esta propiedad son los siguientes:

   - com.sun.jndi.ldap.LdapCtxFactory (para LDAP)
   - com.sun.jndi.fscontext.RefFSContextFactory (para el contexto de sistema de archivos)
   - z/OS com.ibm.jndi.LDAPCtxFactory sólo está soportado en z/OS y proporciona acceso a un servidor LDAP. Sin embargo, esta clase es incompatible con com.sun.jndi.ldap.LdapCtxFactory, ya que los objetos creados con una InitialContextFactory no se pueden leer ni modificar utilizando la otra.

   También puede utilizar la herramienta de administración para conectarse a otros contextos JNDI utilizando tres parámetros definidos en el archivo de configuración JMSAdmin o JMS30Admin . Para utilizar una InitialContextFactory diferente:

   a) Establezca la propiedad **INITIAL_CONTEXT_FACTORY** en el nombre de clase necesario.

   b) Defina el comportamiento de la InitialContextFactory utilizando las propiedades **USE_INITIAL_DIR_CONTEXT**, **NAME_PREFIX** y **NAME_READABILITY_MARKER**.

      Los valores de estas propiedades se describen en los comentarios del archivo de configuración de ejemplo.

   No es necesario que defina las propiedades **USE_INITIAL_DIR_CONTEXT**, **NAME_PREFIX** y **NAME_READABILITY_MARKER** si utiliza uno de los valores de **INITIAL_CONTEXT_FACTORY** soportados. No obstante, puede asignar valores a estas propiedades si desea alterar temporalmente los valores predeterminados del sistema. Por ejemplo, si sus objetos se almacenan en un entorno LDAP, puede cambiar el prefijo predeterminado que la herramienta añade a los nombres de objeto y de contexto, estableciendo la propiedad **NAME_PREFIX** en el prefijo necesario.

   Si omite una o más de las tres propiedades de InitialContextFactory, la herramienta de administración proporciona valores predeterminados adecuados basándose en los valores de las otras propiedades.

2. Defina el URL del contexto inicial de la sesión configurando la propiedad **PROVIDER_URL**.

   Este URL es la raíz de todas las operaciones JNDI realizadas por la herramienta. Se admiten dos formatos de esta propiedad:

   - ldap://hostname/contextname
   - file:[unidad:]/nombre_vía_acceso

   El formato del URL de LDAP puede variar en función del proveedor de LDAP. Consulte la documentación de LDAP para obtener más información.

3. Defina si JNDI pasa credenciales de seguridad al proveedor de servicios configurando la propiedad **SECURITY_AUTHENTICATION**.

   Esta propiedad se utiliza únicamente cuando se utiliza un proveedor de servicios LDAP y puede adoptar uno de tres valores:

   none (autenticación anónima)
   Si establece este parámetro en none, JNDI no pasa ninguna credencial de seguridad al proveedor de servicios y se realiza una *autenticación anónima*.

`simple` (autenticación simple)
Si establece el parámetro en `simple`, se pasan credenciales de seguridad a través de JNDI al proveedor de servicios subyacente. Estas credenciales de seguridad tienen el formato de un nombre distinguido de usuario (DN de usuario) y una contraseña.

`CRAM-MD5` (mecanismo de autenticación CRAM-MD5)
Si establece el parámetro en `CRAM-MD5`, se pasan credenciales de seguridad a través de JNDI al proveedor de servicios subyacente. Estas credenciales de seguridad tienen el formato de un nombre distinguido de usuario (DN de usuario) y una contraseña.

Si no proporciona un valor válido para la propiedad **SECURITY_AUTHENTICATION**, la propiedad adopta de forma predeterminada el valor `none`.

Si se requieren credenciales de seguridad, se le solicitarán cuando se inicialice la herramienta. Puede evitarlo estableciendo las propiedades **PROVIDER_USERDN** y **PROVIDER_PASSWORD** en el archivo de configuración JMSAdmin.

Nota: Si no utiliza estas propiedades, el texto escrito, *incluida la contraseña*, se refleja en la pantalla. Esto puede tener implicaciones de seguridad.

La propia herramienta no realiza ninguna autenticación; la tarea de autenticación se delega al servidor LDAP. El administrador del servidor LDAP debe configurar y mantener los privilegios de acceso a las distintas partes del directorio. Consulte la documentación de LDAP para obtener más información. Si la autenticación falla, la herramienta muestra un mensaje de error apropiado y finaliza.

Puede encontrar información más detallada sobre la seguridad y JNDI en la documentación del sitio web Java de Oracle (Oracle Technology Network for Java Developers).

Inicio de las herramientas JMSAdmin y JMS30Admin

Las herramientas de administración de IBM MQ JMS y Jakarta Messaging tienen una interfaz de línea de mandatos que puede utilizar de forma interactiva o para iniciar un proceso por lotes.

Acerca de esta tarea

La modalidad interactiva proporciona un indicador de mandatos en el que puede entrar mandatos de administración. En la modalidad de proceso por lotes, el mandato para iniciar la herramienta incluye el nombre de un archivo que contiene un script de mandatos de administración.

Procedimiento

Modalidad interactiva

- Para iniciar la herramienta en modalidad interactiva, entre el siguiente mandato:

  JMS 2.0

  ```
  JMSAdmin [-t] [-v] [-cfg config_filename]
  ```

  JM 3.0

  ```
  JMS30Admin [-t] [-v] [-cfg config_filename]
  ```

  donde:

  -t
  Habilita el rastreo (el valor predeterminado es rastreo desactivado).

  El archivo de rastreo se genera en `"%MQ_JAVA_DATA_PATH%"\errors` (Windows) o `/var/mqm/trace` (AIX and Linux). El nombre del archivo de rastreo tiene el formato:

  `mqjms_`*`PID`*`.trc`

donde *PID* es el ID de proceso de la JVM.

-v
Genera una salida detallada (el valor predeterminado es una salida concisa).

-cfg nombre_archivo_config
Indica un archivo de configuración alternativo. Si se omite este parámetro, se utiliza el archivo de configuración predeterminado, `JMSAdmin.config` (para JMS 2.0) o `JMS30Admin.config` (para Jakarta Messaging 3.0). Para obtener más información sobre el archivo de configuración, consulte "Configuración de las herramientas JMSAdmin y JMS30Admin" en la página 723.

Se visualiza un indicador de mandatos, lo que indica que la herramienta está preparada para aceptar mandatos de administración. Este indicador aparece inicialmente como:

```
InitCtx>
```

lo que indica que el contexto actual (es decir, el contexto JNDI al que actualmente hacen referencia todas las operaciones de denominación y directorio) es el contexto inicial definido en el parámetro de configuración **PROVIDER_URL**. Para obtener más información sobre este parámetro, consulte "Configuración de las herramientas JMSAdmin y JMS30Admin" en la página 723.

A medida que se atraviesa el espacio de nombres de directorio, el indicador cambia para reflejarlo, por lo que el indicador siempre muestra el contexto actual.

Modalidad de proceso por lotes

- Para iniciar la herramienta en modalidad de proceso por lotes, entre el siguiente mandato:

  JMS 2.0

  ```
  JMSAdmin test.scp
  ```

  JM 3.0

  ```
  JMS30Admin test.scp
  ```

  donde *test.scp* es un archivo de script que contiene mandatos de administración. Para obtener más información, consulte "Utilización de mandatos de administración con JMSAdmin y JMS30Admin" en la página 726. El último mandato del archivo debe ser el mandato END.

Utilización de mandatos de administración con JMSAdmin y JMS30Admin

Las herramientas de administración de IBM MQ JMS y Jakarta Messaging aceptan mandatos que constan de un verbo de administración y sus parámetros adecuados.

Acerca de esta tarea

La tabla siguiente lista los verbos de administración que puede utilizar al especificar mandatos con las herramientas de administración.

Tabla 38. Verbos de administración

Verbo	Formato abreviado	Descripción
ALTER	ALT	Cambiar al menos una de las propiedades de un objeto administrado
DEFINE	DEF	Crear y almacenar un objeto administrado, o crear un subcontexto
DISPLAY	DIS	Mostrar las propiedades de uno o más objetos administrados almacenados, o el contenido del contexto actual

Tabla 38. Verbos de administración (continuación)		
Verbo	**Formato abreviado**	**Descripción**
DELETE	DEL	Eliminar uno o más objetos administrados del espacio de nombres, o eliminar un subcontexto vacío
CHANGE	CHG	Modificar el contexto actual, permitiendo al usuario desplazarse a cualquier lugar del espacio de nombres de directorio bajo el contexto inicial (pendiente del permiso de seguridad)
COPY	CP	Hacer una copia de un objeto administrado almacenado y almacenarlo con un nombre alternativo
MOVE	MV	Modificar el nombre con el que se ha almacenado un objeto administrado
END		Cerrar la herramienta de administración

Procedimiento

- Si la herramienta de administración aún no se ha iniciado, iníciela tal como se describe en "Inicio de las herramientas JMSAdmin y JMS30Admin" en la página 725.

  Se visualiza el indicador de mandatos, que indica que la herramienta está preparada para aceptar mandatos de administración. Este indicador aparece inicialmente como:

  ```
  InitCtx>
  ```

  Para cambiar el contexto actual, utilice el verbo CHANGE como se describe en "Configurar subcontextos" en la página 727.
- Especifique los mandatos en el siguiente formato:

  ```
  verb [param]*
  ```

  donde **verb** es uno de los verbos de administración listados en Tabla 38 en la página 726. Todos los mandatos válidos contienen un verbo, que aparece al principio del mandato en su forma estándar o abreviada. Los nombres de los verbos no son sensibles a las mayúsculas y minúsculas.
- Para terminar un mandato, pulse Intro, a menos que desee entrar varios mandatos a la vez, en cuyo caso escriba el signo más (+) justo antes de pulsar Intro.

  Normalmente, se pulsa Intro para terminar los mandatos. No obstante, puede alterar temporalmente este comportamiento escribiendo el signo más (+) justo antes de pulsar Intro. Esto le permite entrar mandatos multilínea, tal como se muestra en el siguiente ejemplo:

  ```
  DEFINE Q(BookingsInputQueue) +
  QMGR(QM.POLARIS.TEST) +
  QUEUE(BOOKINGS.INPUT.QUEUE) +
  PORT(1415) +
  CCSID(437)
  ```
- Para cerrar la herramienta de administración, utilice el verbo **END**.

  Este verbo no acepta ningún parámetro.

Configurar subcontextos

Puede utilizar los verbos **CHANGE**, **DEFINE**, **DISPLAY** y **DELETE** para configurar subcontextos de espacio de nombres de directorio.

Acerca de esta tarea

El uso de estos verbos se describe en la siguiente tabla.

Tabla 39. Sintaxis y descripción de los mandatos que se utilizan para manipular subcontextos

Sintaxis del mandato	Descripción
DEFINE CTX(nombreContexto)	Intenta crear un subcontexto hijo del contexto actual, con el nombre nombreContexto. No se ejecuta correctamente si se produce una violación de la seguridad, si el subcontexto ya existe o si el nombre proporcionado no es válido.
DISPLAY CTX	Muestra el contenido del contexto actual. Los objetos administrados se anotan con a, los subcontextos con `[D]`. También se muestra el tipo Java de cada objeto.
DELETE CTX(nombreContexto)	Intenta suprimir el contexto hijo con el nombre nombreContexto del contexto actual. No se ejecuta correctamente si no se encuentra el contexto, si el contexto no está vacío o si se produce una violación de la seguridad.
CHANGE CTX(nombreContexto)	Modifica el contexto actual para que ahora haga referencia al contexto hijo con el nombre nombreContexto. Se puede proporcionar uno de los dos siguientes valores especiales de nombreContexto: **=UP** se traslada al contexto padre del contexto actual **=INIT** se traslada directamente al contexto inicial No se ejecuta correctamente si el contexto especificado no existe o si hay una violación de la seguridad.

Los nombres de los objetos JMS o Jakarta Messaging que se almacenan en un entorno LDAP deben cumplir con los convenios de denominación LDAP. Una de estas convenciones es que los nombres de objeto y de contexto deben incluir un prefijo, como `cn=` (nombre común) u `ou=` (unidad organizativa). La herramienta de administración simplifica el uso de los proveedores de servicio LDAP al permitir hacer referencia a nombres de contexto y de objeto sin un prefijo. Si no se proporciona un prefijo, la herramienta añade automáticamente un prefijo predeterminado al nombre especificado. Para LDAP, este prefijo es `cn=`. Si es necesario, puede cambiar el prefijo predeterminado estableciendo la propiedad **NAME_PREFIX** en el archivo de configuración. Para obtener más información, consulte "Configuración de las herramientas JMSAdmin y JMS30Admin" en la página 723.

Nota: Es posible que tenga que configurar el servidor LDAP para almacenar objetos Java. Para obtener más información, consulte la documentación del servidor LDAP.

crear objetos de JMS

Para crear objetos de destino y fábrica de conexiones JMS o Jakarta Messaging y almacenarlos en un espacio de nombres JNDI, utilice el verbo `DEFINE` . Para almacenar los objetos en un entorno LDAP, debe asignarles nombres que sigan ciertas convenciones. La herramienta de administración puede ayudarle a cumplir las convenciones de denominación LDAP al añadir un prefijo predeterminado a los nombres de objeto.

Acerca de esta tarea

El verbo DEFINE crea un objeto administrado con el tipo, nombre y propiedades que especifique. El nuevo objeto se almacena en el contexto actual.

Los nombres de los objetos JMS o Jakarta Messaging que se almacenan en un entorno LDAP deben cumplir con los convenios de denominación LDAP. Una de estas convenciones es que los nombres de objeto y de contexto deben incluir un prefijo, como cn= (nombre común) u ou= (unidad organizativa). La herramienta de administración simplifica el uso de los proveedores de servicio LDAP al permitir hacer referencia a nombres de contexto y de objeto sin un prefijo. Si no se proporciona un prefijo, la herramienta añade automáticamente un prefijo predeterminado al nombre especificado. Para LDAP, este prefijo es cn=. Si es necesario, puede cambiar el prefijo predeterminado estableciendo la propiedad **NAME_PREFIX** en el archivo de configuración. Para obtener más información, consulte "Configuración de las herramientas JMSAdmin y JMS30Admin" en la página 723.

Nota: Es posible que tenga que configurar el servidor LDAP para almacenar objetos Java. Para obtener más información, consulte la documentación del servidor LDAP.

Procedimiento

1. Si la herramienta de administración aún no se ha iniciado, iníciela tal como se describe en "Inicio de las herramientas JMSAdmin y JMS30Admin" en la página 725.

   Se visualiza el indicador de mandatos, que indica que la herramienta está preparada para aceptar mandatos de administración.
2. Asegúrese de que el indicador de mandatos muestra el contexto en el que desea crear el nuevo objeto.

   Cuando se inicia la herramienta de administración, el indicador aparece inicialmente como:

   ```
   InitCtx>
   ```

   Para cambiar el contexto actual, utilice el verbo CHANGE como se describe en "Configurar subcontextos" en la página 727.
3. Para crear una fábrica de conexiones, un destino de cola o un destino de tema, utilice la siguiente sintaxis de mandato:

   ```
   DEFINE TYPE (name) [property]*
   ```

   Es decir, escriba el verbo DEFINE , seguido de una referencia de objeto administrado *TYPE* (name) , seguido de cero o más propiedades (consulte Propiedades de objetos IBM MQ classes for JMS).
4. Para crear una fábrica de conexiones, un destino de cola o un destino de tema, utilice la siguiente sintaxis de mandato:

   ```
   DEFINE TYPE (name) [property]*
   ```
5. Para visualizar el nuevo objeto creado, utilice el verbo DISPLAY con la siguiente sintaxis de mandato:

   ```
   DISPLAY TYPE (name)
   ```

Ejemplo

El siguiente ejemplo muestra una cola llamada testQueue creada en el contexto inicial utilizando el verbo DEFINE. Puesto que este objeto se va a almacenar en un entorno LDAP, aunque el nombre de objeto `testQueue` no se ha especificado con un prefijo, la herramienta añade uno automáticamente para asegurar el cumplimiento de la convención de denominación LDAP. Someter el mandato DISPLAY `Q(testQueue)` también hace que se añada este prefijo.

```
JM 3.0
InitCtx> DEFINE Q(testQueue)

InitCtx> DISPLAY CTX

Contents of InitCtx
```

```
a cn=testQueue       com.ibm.mq.jakarta.jms.MQQueue

1 Object(s)
0 Context(s)
1 Binding(s), 1 Administered
```

JMS 2.0

```
InitCtx> DEFINE Q(testQueue)

InitCtx> DISPLAY CTX

Contents of InitCtx

a cn=testQueue       com.ibm.mq.jms.MQQueue

1 Object(s)
0 Context(s)
1 Binding(s), 1 Administered
```

Condiciones de error de ejemplo al crear un objeto JMS

Pueden surgir una serie de condiciones de error comunes cuando se crea un objeto.

Ejemplos de estas condiciones de error:

CipherSpec correlacionada con CipherSuite

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) SSLCIPHERSUITE(RC4_MD5_US)
WARNING: Converting CipherSpec RC4_MD5_US to
CipherSuite SSL_RSA_WITH_RC4_128_MD5
```

Propiedad no válida para el objeto

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) PRIORITY(4)
Unable to create a valid object, please check the parameters supplied
Invalid property for a QCF: PRI
```

Tipo no válido para valor de propiedad

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) CCSID(english)
Unable to create a valid object, please check the parameters supplied
Invalid value for CCS property: English
```

Conflicto de propiedades - cliente/enlaces

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) HOSTNAME(polaris.hursley.ibm.com)
Unable to create a valid object, please check the parameters supplied
Invalid property in this context: Client-bindings attribute clash
```

Conflicto de propiedades - Inicialización de salida

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) SECEXITINIT(initStr)
Unable to create a valid object, please check the parameters supplied
Invalid property in this context: ExitInit string supplied
without Exit string
```

Valor de la propiedad fuera del rango válido

```
InitCtx/cn=Trash> DEFINE Q(testQ) PRIORITY(12)
Unable to create a valid object, please check the parameters supplied
Invalid value for PRI property: 12
```

Propiedad desconocida

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) PIZZA(ham and mushroom)
```

```
Unable to create a valid object, please check the parameters supplied
Unknown property: PIZZA
```

Esto son ejemplos de condiciones de error que pueden surgir en Windows al buscar objetos administrados JNDI desde una aplicación JMS.

1. Si está utilizando el proveedor JNDI de WebSphere, com.ibm.websphere.naming.WsnInitialContextFactory, debe utilizar una barra inclinada (/) para acceder a los objetos administrados definidos en subcontextos; por ejemplo, jms/MyQueueName. Si utiliza una barra inclinada invertida (\), se genera una excepción InvalidNameException.
2. Si está utilizando el proveedor JNDI de Oracle, com.sun.jndi.fscontext.RefFSContextFactory, debe utilizar una barra inclinada invertida (\) para acceder a los objetos administrados definidos en subcontextos; por ejemplo, ctx1\\fred. Si utiliza una barra inclinada (/), se genera una excepción NameNotFoundException.

Configurar objetos JMS

Puede utilizar los verbos ALTER, DEFINE, DISPLAY, DELETE, COPY y MOVE para manipular objetos administrados en el espacio de nombres de directorio.

Acerca de esta tarea

En la Tabla 40 en la página 731 se resume el uso de estos verbos. Sustituya *TYPE* por la palabra clave que representa el objeto administrado necesario, tal como se describe en "Configuración de objetos JMS y Jakarta Messaging utilizando las herramientas de administración" en la página 721.

Tabla 40. Sintaxis y descripción de los mandatos que se utilizan para manipular objetos administrados

Sintaxis del mandato	**Descripción**
ALTER *TIPO* (nombre) [propiedad]*	Intenta actualizar las propiedades del objeto administrado con las suministradas. No se ejecuta correctamente si se produce una violación de la seguridad, si no se puede encontrar el objeto especificado o si las nuevas propiedades suministradas no son válidas.
DEFINE *TIPO* (nombre) [propiedad]*	Intenta crear un objeto administrado de tipo *TYPE* con las propiedades proporcionadas y almacenarlo bajo el nombre name en el contexto actual. No se ejecuta correctamente si se produce una violación de la seguridad, si el nombre suministrado no es válido o ya existe un objeto con ese nombre, o si las propiedades suministradas no son válidas.
DISPLAY *TIPO* (nombre)	Muestra las propiedades del objeto administrado de tipo *TYPE* , enlazado bajo el nombre name en el contexto actual. No se ejecuta correctamente si el objeto no existe o si se produce una violación de la seguridad.
DELETE *TIPO* (nombre)	Intenta eliminar el objeto administrado de tipo *TYPE* , que tiene el nombre name, del contexto actual. No se ejecuta correctamente si el objeto no existe o si se produce una violación de la seguridad.
COPY *TIPO* (nombreA) *TIPO* (nombreB)	Realiza una copia del objeto administrado de tipo *TYPE* , con el nombre nameA, nombrando la copia nameB. Todo esto se produce dentro del ámbito del contexto actual. No se ejecuta correctamente si el objeto que se va a copiar no existe, si existe un objeto con el nombre nombreB o si se produce una violación de la seguridad.

Tabla 40. Sintaxis y descripción de los mandatos que se utilizan para manipular objetos administrados (continuación)	
Sintaxis del mandato	**Descripción**
MOVE *TIPO* (nombreA) *TIPO* (nombreB)	Mueve (renombra) el objeto administrado de tipo *TYPE* , que tiene el nombre nameA, a nameB. Todo esto se produce dentro del ámbito del contexto actual. No se ejecuta correctamente si el objeto que se va a mover no existe, si existe un objeto con el nombre nombreB o si se produce una violación de la seguridad.

JMS 2.0 Configurar recursos de JMS 2.0 en WebSphere Application Server

Para configurar recursos de JMS 2.0 en WebSphere Application Server, puede utilizar la consola administrativa o mandatos wsadmin.

Antes de empezar

JM 3.0 Aunque IBM MQ 9.3 y posteriores dan soporte a Jakarta Messaging 3.0, WebSphere Application Server no tiene actualmente un soporte equivalente. Por lo tanto, en WebSphere Application Server, configure los recursos de Java Message Service 2.0 .

Acerca de esta tarea

Las aplicaciones de Java Message Service 2.0 normalmente se basan en objetos configurados externamente que describen cómo se conecta la aplicación a su proveedor de JMS y los destinos a los que accede. Las aplicaciones JMS utilizan Java Naming Directory Interface (JNDI) para acceder a los siguientes tipos de objeto en tiempo de ejecución:

- Especificaciones de activación (utilizadas por los servidores de aplicaciones de Java EE)
- Fábricas de conexiones unificadas (con JMS 1.1 y posteriores, las fábricas de conexiones independientes del dominio (unificadas) se prefieren a las fábricas de conexiones de cola específicas del dominio y a las fábricas de conexiones de tema)
- Fábricas de conexiones de tema (utilizadas por aplicaciones JMS 1.0)
- Fábricas de conexiones de cola (utilizadas por aplicaciones JMS 1.0)
- Colas
- Temas

A través del proveedor de mensajería de IBM MQ en WebSphere Application Server, las aplicaciones de mensajería de Java Message Service (JMS) pueden utilizar el sistema IBM MQ como proveedor externo de recursos de mensajería de JMS . Para hacer posible este enfoque, debe configurar el proveedor de mensajería IBM MQ en WebSphere Application Server para definir los recursos de JMS para conectarse a cualquier gestor de colas en la red de IBM MQ.

Puede utilizar WebSphere Application Server para configurar recursos de IBM MQ para aplicaciones (por ejemplo, fábricas de conexiones de cola) y para gestionar mensajes y suscripciones asociados a destinos JMS. La seguridad se administra mediante IBM MQ.

Tareas relacionadas

Utilización de IBM MQ y WebSphere Application Server juntos

Temas de WebSphere Application Server

Interoperación mediante el proveedor de mensajería de IBM MQ

Gestión de la mensajería con el proveedor de mensajería de IBM MQ

Correlación de los nombres del panel de la consola administrativa con nombres de mandatos y nombres de IBM MQ

JMS 2.0 Configurar recursos de JMS 2.0 utilizando la consola administrativa

Puede utilizar la consola administrativa de WebSphere Application Server para configurar especificaciones de activación, fábricas de conexiones y destinos para el proveedor de IBM MQ JMS.

Acerca de esta tarea

Puede utilizar la consola administrativa de WebSphere Application Server para crear, ver o modificar cualquiera de los siguientes recursos:

- Especificaciones de activación
- Fábricas de conexiones independientes del dominio (JMS 1.1 o posterior)
- Fábricas de conexiones de cola
- Fábricas de conexiones de tema
- Colas
- Temas

Los pasos siguientes proporcionan una visión general de las formas en que puede utilizar la consola administrativa para configurar recursos de JMS para utilizarlos con el proveedor de mensajería IBM MQ. Cada paso incluye el nombre del tema de la documentación del producto WebSphere Application Server que puede consultar para obtener más información. Consulte los *Enlaces relacionados* para ver enlaces a estos temas en IBM Documentation.

En una célula de WebSphere Application Server de una versiones mixtas, puede administrar los recursos de IBM MQ en nodos de todas las versiones. No obstante, algunas propiedades no están disponibles en todas las versiones. En esta situación, sólo las propiedades de ese nodo en concreto se muestran en la consola administrativa.

Procedimiento

Para crear o configurar una especificación de activación para utilizarla con el proveedor de mensajería IBM MQ:

- Para crear una especificación de activación, utilice al asistente Crear recurso IBM MQ JMS.

  Puede utilizar el asistente para especificar todos los detalles de la especificación de activación, o puede optar por especificar los detalles de conexión para IBM MQ utilizando una tabla de definiciones de canal de cliente (CCDT). Cuando especifique los detalles de conexión utilizando el asistente, puede elegir especificar la información de host y de puerto por separado o, si está utilizando un gestor de colas multiinstancia, especificar la información de host y de puerto en forma de una lista de nombres de conexión. Para obtener más información, consulte *Creación de una especificación de activación para el proveedor de mensajería de IBM MQ*.
- Para ver o cambiar las propiedades de configuración de una especificación de activación, utilice el panel de valores de fábrica de conexiones de proveedor de mensajería de IBM MQ de la consola administrativa.

  Estas propiedades de configuración controlan cómo se crean las conexiones con las colas y temas asociados. Para obtener más información, consulte *Configuración de una especificación de activación para el proveedor de mensajería de IBM MQ*.

Para crear o configurar una fábrica de conexiones unificada, una fábrica de conexiones de cola o una fábrica de conexiones de tema para utilizarla con el proveedor de mensajería de IBM MQ:

- Para crear una fábrica de conexiones, primero seleccione el tipo de fábrica de conexiones que desea crear y, a continuación, utilice el asistente de creación de recursos de IBM MQ JMS para especificar los detalles.
  - Si su aplicación JMS va a utilizar sólo la mensajería punto a punto, cree una fábrica de conexiones específica de dominio para el dominio de mensajería punto a punto que se puede utilizar para crear conexiones específicamente para la mensajería punto a punto.

- Si su aplicación JMS va a utilizar sólo la mensajería de publicación/suscripción, cree una fábrica de conexiones específica de dominio para el dominio de mensajería de publicación/suscripción que se puede utilizar para crear conexiones específicamente para la mensajería de publicación/suscripción.
- Para JMS 1.1 o posterior, cree una fábrica de conexiones independiente del dominio que se puede utilizar tanto para la mensajería punto a punto como para la mensajería de publicación/suscripción, y que permite a la aplicación realizar trabajo punto a punto y de publicación/suscripción bajo la misma transacción.

Puede decidir utilizar el asistente para especificar todos los detalles de la fábrica de conexiones, o puede decidir especificar los detalles de conexión para IBM MQ utilizando una tabla de definiciones de canal de cliente (CCDT). Cuando especifique los detalles de conexión utilizando el asistente, puede elegir especificar la información de host y de puerto por separado o, si está utilizando un gestor de colas multiinstancia, especificar la información de host y de puerto en forma de una lista de nombres de conexión. Para obtener más información, consulte *Creación de una fábrica de conexiones para el proveedor de mensajería de IBM MQ*.

Para ver o cambiar las propiedades de configuración de una fábrica de conexiones:

- Utilice el panel de valores de fábrica de conexiones de la consola administrativa correspondiente al tipo de fábrica de conexiones que desea configurar.

  Las propiedades de configuración controlan cómo se crean las conexiones con las colas y temas asociados. Para obtener más información, consulte *Configuración de una fábrica de conexiones para el proveedor de mensajería de IBM MQ* o *Configuración de una fábrica de conexiones de cola para el proveedor de mensajería de IBM MQ* o *Configuración de una fábrica de conexiones de tema para el proveedor de mensajería de IBM MQ*.

Para configurar un destino de cola JMS para la mensajería punto a punto con el proveedor de mensajería IBM MQ:

- Utilice el panel de valores de cola de proveedor de mensajería IBM MQ de la consola administrativa para definir los siguientes tipos de propiedades:
  - Propiedades generales, incluyendo propiedades de administración y propiedades de cola de IBM MQ.
  - Propiedades de conexión que especifican cómo conectarse al gestor de colas que aloja la cola.
  - Propiedades avanzadas que controlan el comportamiento de las conexiones realizadas a destinos del proveedor de mensajería IBM MQ.
  - Cualquier propiedad personalizad para el destino de cola.

  Para obtener más información, consulte *Configuración de una cola para el proveedor de mensajería de IBM MQ*.

Para crear o configurar un destino de tema JMS para la mensajería de publicación/suscripción con el proveedor de mensajería IBM MQ:

- Utilice el panel de valores de tema de proveedor de mensajería IBM MQ para definir los siguientes tipos de propiedades:
  - Propiedades generales, incluyendo propiedades de administración y propiedades de tema de IBM MQ.
  - Propiedades avanzadas que controlan el comportamiento de las conexiones realizadas a destinos del proveedor de mensajería IBM MQ.
  - Cualquier propiedad personalizad para el destino de cola.

  Para obtener más información, consulte *Configuración de un tema para el proveedor de mensajería de IBM MQ*.

Conceptos relacionados

"Gestores de colas multiinstancia" en la página 530

Los gestores de colas multiinstancia son instancias del mismo gestor de cola configuradas en diferentes servidores. Una instancia del gestor de colas se define como la instancia activa y otra instancia se define

como la instancia en espera. Si la instancia activa falla, el gestor de colas multiinstancia se reinicia automáticamente en el servidor en espera.

Tareas relacionadas

“Configuración de una tabla de definición de canal de cliente en formato binario” en la página 45
La tabla de definición de canal de cliente (CCDT) determina las definiciones de canal y la información de autenticación que utilizan las aplicaciones cliente para poder conectarse al gestor de colas. En Multiplataforms, se crea automáticamente una tabla de definición de canal de cliente binaria que contiene valores predeterminados cuando se crea el gestor de colas. Puede utilizar el mandato **runmqsc** para actualizar una tabla de definición de canal de cliente binaria.

“Configurar la mensajería de publicación/suscripción” en la página 455
Puede iniciar, detener y visualizar el estado de la publicación/suscripción en cola. También puede añadir y eliminar corrientes de datos, y añadir y suprimir gestores de colas de una jerarquía de intermediarios.

Temas de WebSphere Application Server

Especificaciones de activación del proveedor de mensajería de IBM MQ
Creación de una especificación de activación para el proveedor de mensajería de IBM MQ
Configuración de una especificación de activación para el proveedor de mensajería de IBM MQ
Creación de una fábrica de conexiones para el proveedor de mensajería de IBM MQ
Configuración de una fábrica de conexiones unificada para el proveedor de mensajería de IBM MQ
Configuración de una fábrica de conexiones de cola para el proveedor de mensajería de IBM MQ
Configuración de una fábrica de conexiones de tema para el proveedor de mensajería de IBM MQ
Configuración de una cola para el proveedor de mensajería de IBM MQ
Configuración de un tema para el proveedor de mensajería de IBM MQ

JMS 2.0 Configurar recursos de JMS 2.0 utilizando mandatos de script wsadmin

Puede utilizar mandatos de script wsadmin de WebSphere Application Server para crear, modificar, suprimir o mostrar información sobre especificaciones de activación, fábricas de conexiones, colas y temas JMS. También puede visualizar y gestionar los valores para el adaptador de recursos de IBM MQ.

Acerca de esta tarea

Los pasos siguientes proporcionan una visión general de las formas en que puede utilizar los mandatos wsadmin de WebSphere Application Server para configurar recursos de JMS para utilizarlos con el proveedor de mensajería IBM MQ. Para obtener más información sobre cómo utilizar estos mandatos, consulte *Enlaces relacionados* para ver los enlaces a la documentación del producto de WebSphere Application Server.

Para ejecutar un mandato, utilice el objeto AdminTask del cliente de scripts wsadmin.

Después de utilizar un mandato para crear un nuevo objeto o realizar cambios, guarde los cambios en la configuración maestra. Por ejemplo, utilice el siguiente mandato:

```
AdminConfig.save()
```

Para ver una lista de los mandatos administrativos del proveedor de mensajería IBM MQ disponibles, con una breve descripción de cada mandato, entre el siguiente mandato en el indicador de wsadmin:

```
print AdminTask.help('WMQAdminCommands')
```

Para ver ayuda general sobre un mandato determinado, entre el siguiente mandato en el indicador de wsadmin:

```
print AdminTask.help('command_name')
```

Procedimiento

Para listar todos los recursos del proveedor de mensajería IBM MQ definidos en el ámbito en el que se emite un mandato, utilice los siguientes mandatos.

- Para listar las especificaciones de activación, utilice el mandato **listWMQActivationSpecs**.
- Para listar las fábricas de conexiones, utilice el mandato **listWMQConnectionFactories**.
- Para listar los destinos de tipo cola, utilice el mandato **listWMQQueues**.
- Para listar los destinos de tipo tema, utilice el mandato **listWMQTopics**.

Para crear un recurso JMS para el proveedor de mensajería IBM MQ en un ámbito específico, utilice los siguientes mandatos.

- Para crear una especificación de activación, utilice el mandato **createWMQActivationSpec**.

  Puede crear una especificación de activación especificando todos los parámetros que se utilizarán para establecer una conexión, o puede crear la especificación de activación para que utilice una tabla de definiciones de canal de cliente (CCDT) para localizar el gestor de colas al que conectarse.
- Para crear una fábrica de conexiones, utilice el mandato **createWMQConnectionFactory**, con el parámetro **-type** para especificar el tipo de fábrica de conexiones que desea crear:
  - Si su aplicación JMS va a utilizar sólo la mensajería punto a punto, cree una fábrica de conexiones específica de dominio para el dominio de mensajería punto a punto que se puede utilizar para crear conexiones específicamente para la mensajería punto a punto.
  - Si su aplicación JMS va a utilizar sólo la mensajería de publicación/suscripción, cree una fábrica de conexiones específica de dominio para el dominio de mensajería de publicación/suscripción que se puede utilizar para crear conexiones específicamente para la mensajería de publicación/suscripción.
  - Para JMS 1.1 o posterior, cree una fábrica de conexiones independiente del dominio que se puede utilizar tanto para la mensajería punto a punto como para la mensajería de publicación/suscripción, y que permite a la aplicación realizar trabajo punto a punto y de publicación/suscripción bajo la misma transacción.

  El tipo predeterminado es una fábrica de conexiones independiente del dominio.
- Para crear un destino de tipo cola, utilice el mandato **createWMQQueue**.
- Para crear un destino de tipo tema, utilice el mandato **createWMQTopic**.

Para modificar un recurso JMS para el proveedor de mensajería IBM MQ en un ámbito específico, utilice los siguientes mandatos.

- Para modificar una especificación de activación, utilice el mandato **modifyWMQActivationSpec**.

  No puede cambiar el tipo de una especificación de activación. Por ejemplo, no puede crear una especificación de activación en la que se especifique toda la información de configuración manualmente y luego modificarla para que utilice una CCDT.
- Para modificar una fábrica de conexiones, utilice el mandato **modifyWMQConnectionFactory**.
- Para modificar un destino de tipo cola, utilice el mandato **modifyWMQQueue**.
- Para modificar un destino de tipo tema, utilice el mandato **modifyWMQTopic**.

Para suprimir un recurso JMS para el proveedor de mensajería IBM MQ en un ámbito específico, utilice los siguientes mandatos.

- Para suprimir una especificación de activación, utilice el mandato **deleteWMQActivationSpec**.
- Para suprimir una fábrica de conexiones, utilice el mandato **deleteWMQConnectionFactory**.
- Para suprimir un destino de tipo cola, utilice el mandato **deleteWMQQueue**.
- Para suprimir un destino de tipo tema, utilice el mandato **deleteWMQTopic**.

Para visualizar información sobre un recurso específico del proveedor de mensajería IBM MQ, utilice los siguientes mandatos.

- Para visualizar todos los parámetros, y sus valores, asociados a una especificación de activación determinada, utilice el mandato **showWMQActivationSpec**.

- Para visualizar todos los parámetros y sus valores, asociados a una fábrica de conexiones determinada, utilice el mandato **showWMQConnectionFactory**.
- Para visualizar todos los parámetros, y sus valores, asociados a un destino de tipo cola determinado, utilice el mandato **showWMQQueue**.
- Para visualizar todos los parámetros, y sus valores, asociados a un destino de tipo de tema, utilice el mandato **deleteWMQTopic** .

Para gestionar los valores para el adaptador de recursos de IBM MQ o el proveedor de mensajería IBM MQ, utilice los siguientes mandatos.

- Para gestionar los valores del adaptador de recursos de IBM MQ que está instalado en un ámbito específico, utilice el mandato **manageWMQ**.
- Para visualizar todos los parámetros, y sus valores, que se pueden establecer mediante el mandato **manageWMQ**, utilice el mandato **showWMQ**. Estos valores están relacionados con el adaptador de recursos de IBM MQ o el proveedor de mensajería de IBM MQ. El mandato **showWMQ** también muestra las propiedades personalizadas establecidas el adaptador de recursos de IBM MQ.

Conceptos relacionados

"Gestores de colas multiinstancia" en la página 530
Los gestores de colas multiinstancia son instancias del mismo gestor de cola configuradas en diferentes servidores. Una instancia del gestor de colas se define como la instancia activa y otra instancia se define como la instancia en espera. Si la instancia activa falla, el gestor de colas multiinstancia se reinicia automáticamente en el servidor en espera.

Tareas relacionadas

"Configuración de una tabla de definición de canal de cliente en formato binario" en la página 45
La tabla de definición de canal de cliente (CCDT) determina las definiciones de canal y la información de autenticación que utilizan las aplicaciones cliente para poder conectarse al gestor de colas. En Multiplataforms, se crea automáticamente una tabla de definición de canal de cliente binaria que contiene valores predeterminados cuando se crea el gestor de colas. Puede utilizar el mandato **runmqsc** para actualizar una tabla de definición de canal de cliente binaria.

"Configurar la mensajería de publicación/suscripción" en la página 455
Puede iniciar, detener y visualizar el estado de la publicación/suscripción en cola. También puede añadir y eliminar corrientes de datos, y añadir y suprimir gestores de colas de una jerarquía de intermediarios.

Temas de WebSphere Application Server

Mandato **createWMQActivationSpec**
Mandato **createWMQConnectionFactory**
Mandato **createWMQQueue**
Mandato **createWMQTopic**
Mandato **deleteWMQActivationSpec**
Mandato **deleteWMQConnectionFactory**
Mandato **deleteWMQQueue**
Mandato **deleteWMQTopic**
Mandato **listWMQActivationSpecs**
Mandato **listWMQConnectionFactories**
Mandato **listWMQQueues**
Mandato **listWMQTopics**
Mandato **modifyWMQActivationSpec**
Mandato **modifyWMQConnectionFactory**
Mandato **modifyWMQQueue**
Mandato **modifyWMQTopic**
Mandato **showWMQActivationSpec**
Mandato **showWMQConnectionFactory**
Mandato **showWMQQueue**
Mandato **showWMQTopic**

Mandato **showWMQ**
Mandato **manageWMQ**

JMS 2.0 Utilización de suscripciones compartidas de JMS 2.0

En WebSphere Application Server traditional 9.0, puede configurar y utilizar suscripciones compartidas de JMS 2.0 con IBM MQ 9.0.

Acerca de esta tarea

La especificación JMS 2.0 introducía el concepto de suscripciones compartidas, que permite que uno o varios consumidores abran una única suscripción. Los mensajes se comparten entre todos estos consumidores. No hay ninguna restricción respecto al lugar donde estén estos consumidores a condición de que se conecten al mismo gestor de colas.

Las suscripciones compartidas puede ser duraderas o no duraderas, con la misma semántica que lo que ahora se conoce como suscripciones no compartidas.

Para que un consumidor pueda identificar qué suscripción debe utilizar, debe proporcionar un nombre de suscripción. Esto es similar a las suscripciones duraderas no compartidas, pero se necesita un nombre de suscripción en todos los casos en que es necesaria una suscripción compartida. Sin embargo, no es necesario un ID de cliente en el caso de una suscripción compartida duradera; se puede proporcionar uno pero no es obligatorio.

Mientras que las suscripciones compartidas pueden considerarse como un mecanismo de equilibrio de carga, ni en IBM MQ ni en la especificación JMS 2.0 hay ningún compromiso sobre cómo se distribuyen los mensajes entre los consumidores.

En WebSphere Application Server traditional 9.0 hay un adaptador de recursos de IBM MQ 9.0 instalado previamente.

Los pasos siguientes muestran cómo configurar una especificación de activación para utilizar una suscripción compartida duradera o compartida no duradera utilizando la consola administrativa de WebSphere Application Server traditional.

Procedimiento

Primero cree los objetos en JNDI.

1. Cree un destino de tema en JNDI como es habitual (consulte “Configurar recursos de JMS 2.0 utilizando la consola administrativa” en la página 733).
2. Cree una especificación de activación (consulte “Configurar recursos de JMS 2.0 utilizando la consola administrativa” en la página 733).

   Puede crear la especificación de activación con exactamente las propiedades que necesita. Si desea utilizar una suscripción duradera, puede seleccionarla durante la creación y especificar un nombre. Si desea utilizar una suscripción no duradera, no puede especificar un nombre en este punto. En lugar de ello, debe crear una propiedad personalizada para el nombre de suscripción.

Actualice la especificación de activación que ha creado con las propiedades personalizadas necesarias. Hay dos propiedades personalizadas que es posible que necesite especificar:

- En todos los casos, debe crear una propiedad personalizada para especificar que esta especificación de activación debe utilizar una suscripción compartida.
- Si la suscripción se ha creado como no duradera, la propiedad de nombre de suscripción debe establecerse como una propiedad personalizada.

La tabla siguiente muestra el valor válido que puede especificar para cada propiedad personalizada:

Nombre de propiedad	Tipo	Valores válidos
sharedSubscription	Serie	true, false

Nombre de propiedad	Tipo	Valores válidos
subscriptionName	Serie	Serie java de longitud distinta de cero

3. Seleccione la especificación de activación en la lista visualizada en el formulario de **Colección de especificaciones de activación**.

   Los detalles de la especificación de activación se muestran en el formulario **Valores de especificación de activación de proveedor de mensajería de IBM MQ**.
4. En el formulario **Valores de especificación de activación de proveedor de mensajería de IBM MQ**, pulse **Propiedades personalizadas**.

   Se visualiza el formulario **Propiedades personalizadas**.
5. Si está utilizando una suscripción duradera, cree la propiedad personalizada subscriptionName.

   En el panel **Propiedades personalizadas** de la especificación de activación, pulse **Nuevo**y, a continuación, especifique los siguientes detalles:

   Nombre

   Nombre de la propiedad personalizada, que en este caso es `subscriptionName`.

   Valor

   Valor de la propiedad personalizada. Puede utilizar los nombres de JNDI en el campo **Valor**, por ejemplo `WASSharedSub0ne`.

   Tipo

   Tipo de la propiedad personalizada. Seleccione el tipo de propiedad personalizada de la lista, que en este caso debe ser `java.lang.String`.
6. Para la suscripción compartida duradera y la suscripción compartida no duradera, cree la propiedad personalizada sharedSubscription.

   En el panel **Propiedades personalizadas** de la especificación de activación, pulse **Nuevo**y, a continuación, especifique los siguientes detalles:

   Nombre

   Nombre de la propiedad personalizada, que en este caso es `sharedSubscription`.

   Valor

   Valor de la propiedad personalizada. Para especificar que la especificación de activación utilice una suscripción compartida, establezca el valor en `true`. Si más adelante desea dejar de utilizar una suscripción compartida para esta especificación de activación, puede hacerlo estableciendo el valor de esta propiedad personalizada en `false`.

   Tipo

   Tipo de la propiedad personalizada. Seleccione el tipo de propiedad personalizada de la lista, que en este caso debe ser `java.lang.String`.
7. Cuando se hayan establecido las propiedades, reinicie el servidor de aplicaciones.

   Los beans controlados por mensaje (MDB) para las especificaciones de activación se controlan entonces cuando llegan los mensajes, pero sólo los MDB comparten los mensajes que se envían.

Conceptos relacionados

Suscripciones clonadas y compartidas

Durabilidad de suscripción

Tareas relacionadas

Configuración del adaptador de recursos para la comunicación de entrada

Información relacionada para WebSphere Application Server traditional 9.0

Configuración de un tema para el proveedor de mensajería de IBM MQ

Especificaciones de activación del proveedor de mensajería de IBM MQ

Creación de una especificación de activación para el proveedor de mensajería de IBM MQ
Configuración de una especificación de activación para el proveedor de mensajería de IBM MQ
Configuración de propiedades personalizadas para recursos del IBM MQ proveedor de mensajería JMS

JMS 2.0 Utilización de propiedades de búsqueda de destino y fábrica de conexiones de JMS 2.0

En WebSphere Application Server traditional 9.0, las propiedades ConnectionFactoryLookup y DestinationLookup de una especificación de activación se pueden proporcionar con un nombre de JNDI de un objeto administrado que se debe utilizar con preferencia respecto a otras propiedades de especificación de activación.

Acerca de esta tarea

La especificación JMS 2.0 especifica dos propiedades adicionales en la especificación de activación utilizada para controlar los beans controlados por mensajes (MDB). Anteriormente, cada proveedor tenía que especificar propiedades personalizadas en la especificación de activación para proporcionar los detalles necesarios para conectarse a un sistema de mensajería y definir de qué destino se obtienen los mensajes.

Las propiedades connectionFactoryLookup y destinationLookup ahora estándares se pueden utilizar para proporcionar un nombre de JNDI del objeto relevante que se debe buscar y utilizar. En WebSphere Application Server traditional 9.0 hay un adaptador de recursos de IBM MQ 9.0 instalado previamente.

Los siguientes pasos muestran cómo personalizar y utilizar estas dos propiedades utilizando la consola administrativa de WebSphere Application Server traditional.

Procedimiento

Primero cree los objetos en JNDI.

1. Cree la ConnectionFactory en JNDI como es habitual (consulte “Configurar recursos de JMS 2.0 utilizando la consola administrativa” en la página 733).
2. Cree el destino en JNDI como es habitual (consulte “Configurar recursos de JMS 2.0 utilizando la consola administrativa” en la página 733).

   El objeto de destino debe tener los valores correctos.
3. Cree la especificación de activación utilizando los valores que se necesitan (consulte “Configurar recursos de JMS 2.0 utilizando la consola administrativa” en la página 733).

   Puede crear la especificación de activación con exactamente las propiedades que necesita. Sin embargo, debe tener en cuenta los puntos siguientes:

   - Si desea que el adaptador de recursos de IBM MQ utilice las propiedades de búsqueda de destino y la fábrica de conexiones de Java EE, es menos relevante qué propiedades se utilizan cuando se crea la especificación de activación (consulte Propiedades de ActivationSpec ConnectionFactoryLookup y DestinationLookup .
   - Sin embargo, cualquier propiedad que aún no esté definido en la fábrica de conexiones o el destino se debe especificar de todas formas en la especificación de activación. Por lo tanto, debe definir las propiedades de consumidor de conexión y propiedades adicionales así como la información de autenticación que se utiliza cuando una conexión se crea realmente.
   - De las propiedades que se definen en la fábrica de conexiones, la propiedad IdCliente tiene un proceso especial. Esto es porque es un escenario común utilizar una única fábrica de conexiones con varias especificaciones de activación. Aunque esto simplifica la administración, la especificación de JMS requiere ID de cliente exclusivos, por lo tanto la especificación de activación necesita tener la posibilidad de alterar temporalmente cualquier valor establecido en ConnectionFactory. Si no se establece ningún ID de cliente en la especificación de activación, se utiliza cualquier valor de la fábrica de conexiones.

Actualice la especificación de activación que ha creado con las dos nuevas propiedades personalizadas utilizando la consola administrativa de WebSphere Application Server como se describe en el paso “4” en la página 741 o, en su lugar, utilice anotaciones como se describe en el paso “5” en la página 741.

4. Actualice la especificación de activación en la consola administrativa de WebSphere Application Server.

   Estas dos propiedades deben establecerse en el panel de propiedades personalizadas de la especificación de activación. Estas propiedades no están presentes en los paneles de especificación de activación principales o en el asistente de creación de especificación de activación.

   a) Seleccione la especificación de activación en la lista visualizada en el formulario de **Colección de especificaciones de activación**.

      Los detalles de la especificación de activación se muestran en el formulario **Valores de especificación de activación de proveedor de mensajería de IBM MQ**.

   b) En el formulario **Valores de especificación de activación de proveedor de mensajería de IBM MQ**, pulse **Propiedades personalizadas**.

      Se visualiza el formulario **Propiedades personalizadas**.

   c) En el formulario **Propiedades personalizadas**, cree dos nuevas propiedades personalizadas, ambas de tipo java.lang.String.

      En cada caso, pulse **Nuevo** y especifique los siguientes detalles para la propiedad personalizada:

      Nombre

      Nombre de la propiedad personalizada, `connectionFactoryLookup` o `destinationLookup`.

      Valor

      Valor de la propiedad personalizada. Puede utilizar los nombres de JNDI el campo **Valor**, por ejemplo `QuoteCF` y `QuoteQ`.

      Tipo

      Tipo de la propiedad personalizada. Seleccione el tipo de propiedad personalizada de la lista, que en este caso debe ser `java.lang.String`.

   El MDB desplegado ahora utilizará estos valores para crear la fábrica de conexiones y el destino. Al desplegar el MDB, no hay ningún requisito para establecer la configuración de valores de JNDI.

5. Utilice anotaciones en lugar de la especificación de activación.

   También es posible utilizar anotaciones en el código de MDB para especificar valores. Por ejemplo, utilizando los nombres de JNDI `QuoteCF` y `QuoteQ`, este es el aspecto que tendría el código:

```
@MessageDriven(activationConfig = {
        @ActivationConfigProperty(propertyName =  "destinationType" , propertyValue =
"javax.jms.Topic" ),
        @ActivationConfigProperty(propertyName =  "destinationLookup" , propertyValue =
"QuoteQ" ),
        @ActivationConfigProperty(propertyName =  "connectionFactoryLookup" , propertyValue
=  "QuoteCF" )}, mappedName =  "LookupMDB" )
        @TransactionAttribute(TransactionAttributeType.REQUIRED)
        @TransactionManagement(TransactionManagementType.CONTAINER)
        publicclass LookupMDB implements MessageListener {
```

Tareas relacionadas

Configuración del adaptador de recursos para la comunicación de entrada

Información relacionada para WebSphere Application Server traditional 9.0

Configuración de una fábrica de conexiones unificada para el proveedor de mensajería de IBM MQ

Configuración de un tema para el proveedor de mensajería de IBM MQ

Especificaciones de activación del proveedor de mensajería de IBM MQ

Creación de una especificación de activación para el proveedor de mensajería de IBM MQ

Configuración de una especificación de activación para el proveedor de mensajería de IBM MQ

Configuración de propiedades personalizadas para recursos del IBM MQ proveedor de mensajería JMS

Configuración de WebSphere Application Server para utilizar el último nivel de mantenimiento del adaptador de recursos

Para asegurar que el adaptador de recursos de IBM MQ se actualice automáticamente al último nivel de mantenimiento disponible cuando aplique fixpacks de WebSphere Application Server, puede configurar todos los servidores del entorno para que utilicen la versión más reciente del adaptador de recursos incluida en el fixpack de WebSphere Application Server que ha aplicado a la instalación de cada nodo.

Antes de empezar

Importante:

- JM 3.0 WebSphere Application Server traditional no da soporte actualmente a Jakarta EE. Consulte Declaración de soporte del adaptador de recursos deIBM MQ.
- Si está utilizando WebSphere Application Server 8.5 o anterior en cualquier plataforma, no instale el adaptador de recursos IBM MQ 8.0 o posterior en el servidor de aplicaciones. El adaptador de recursos IBM MQ 8.0 o posterior solo se puede desplegar en un servidor de aplicaciones que admita JMS 2.0. Sin embargo, WebSphere Application Server 8.5 o anterior solo admite JMS 1.1.

Acerca de esta tarea

Utilice esta tarea si alguna de las siguientes circunstancias se aplica a su configuración, y desea configurar todos los servidores del entorno para que utilicen la versión más reciente del adaptador de recursos de IBM MQ:

- Los registros de JVM de cualquier servidor de aplicaciones del entorno muestran la siguiente información de versión del adaptador de recursos de IBM MQ después de haber aplicado WebSphere Application Server 7.0.0 Fix Pack 1 o posterior:

  ```
  WMSG1703I: Implementación de RAR Versión 7.0.0.0-k700-L080820
  ```

- Los registros de JVM de cualquier servidor de aplicaciones del entorno contienen la siguiente entrada:

  ```
  WMSG1625E: No fue posible detectar
  el código del proveedor de mensajería de IBM MQ en la vía de acceso especificada < null>
  ```

- Uno o más nodos se han actualizado manualmente con anterioridad para utilizar un nivel de mantenimiento específico del adaptador de recursos de IBM MQ que ahora ha sido reemplazado por la versión más reciente del adaptador de recursos contenida en el nivel de mantenimiento de WebSphere Application Server actual.

El directorio *profile_root* al que hacen referencia los ejemplos es el directorio de inicio del perfil WebSphere Application Server , por ejemplo, `C:\Program Files\IBM\WebSphere\AppServer1`.

Cuando haya realizado los pasos siguientes para todas las células y las instalaciones de un solo servidor en el entorno, los servidores reciben automáticamente mantenimiento para el adaptador de recursos de IBM MQ cuando se aplica un nuevo fixpack de WebSphere Application Server.

Procedimiento

1. Inicie el servidor de aplicaciones. Si el perfil forma parte de una configuración de despliegue de red, inicie el gestor de despliegue y todos los agentes de nodo. Si el perfil contiene un agente administrativo, inicie el agente administrativo.
2. Compruebe el nivel de mantenimiento del adaptador de recursos de IBM MQ.
   a) Abra una ventana de indicador de mandatos y vaya al directorio *profile_root*`\bin` .
   Por ejemplo, especifique `cd C:\Program Files\IBM\WebSphere\AppServer1\bin`.
   b) Inicie la herramienta wsadmin especificando `wsadmin.bat -lang jython`y, a continuación, si se le solicita, especifique el nombre de usuario y la contraseña.
   c) Escriba el siguiente mandato y luego pulse Intro dos veces.

```
wmqInfoMBeansUnsplit = AdminControl.queryNames("WebSphere:type=WMQInfo,*")
wmqInfoMBeansSplit = AdminUtilities.convertToList(wmqInfoMBeansUnsplit)
for wmqInfoMBean in wmqInfoMBeansSplit: print wmqInfoMBean; print AdminControl.invoke(wmqInfoMBean,
'getInfo', '')
```

También puede ejecutar este mandato en Jacl. Para obtener más información sobre cómo hacerlo, consulte *Asegurarse de que los servidores utilizan el último nivel de mantenimiento de adaptador de recursos IBM MQ disponible* en la documentación del producto WebSphere Application Server.

d) Busque el mensaje `WMSG1703I` en la salida visualizada del mandato y compruebe el nivel del adaptador de recursos.

Por ejemplo, en WebSphere Application Server 7.0.1 Fix Pack 5, el mensaje debería ser:

```
WMSG1703I: Implementación de RAR Versión 7.0.1.3-k701-103-100812
```

Este mensaje muestra que la versión es 7.0.1.3-k701-103-100812, que es el nivel correcto del adaptador de recursos para este fixpack. Sin embargo, si se visualiza en su lugar el mensaje siguiente, significa que debe ajustar el adaptador de recursos al nivel de mantenimiento correcto para WebSphere Application Server 7.0.1 Fix Pack 5.

```
WMSG1703I: Implementación de RAR Versión 7.0.0.0-k700-L080820
```

3. Copie el siguiente script Jython en un archivo denominado `convertWMQRA.pyy`, a continuación, guárdelo en el directorio raíz del perfil, por ejemplo, `C:\Program Files\IBM\WebSphere\AppServer1\bin`.

```
ras = AdminUtilities.convertToList(AdminConfig.list('J2CResourceAdapter'))

for ra in ras :
  desc = AdminConfig.showAttribute(ra, "description")
  if (desc == "WAS 7.0 Built In MQ Resource Adapter") or (desc == "WAS 7.0.0.1 Built In MQ Resource
Adapter"):
    print "Updating archivePath and classpath of " + ra
    AdminConfig.modify(ra, [['archivePath', "${WAS_INSTALL_ROOT}/installedConnectors/wmq.jmsra.rar"]])
    AdminConfig.unsetAttributes(ra, ['classpath'])
    AdminConfig.modify(ra, [['classpath', "${WAS_INSTALL_ROOT}/installedConnectors/wmq.jmsra.rar"]])
    AdminConfig.save()
  #end if
#end for
```

Consejo: Al guardar el archivo, asegúrese de guardarlo como archivo python en vez de como archivo de texto.

4. Utilice la herramienta wsadmin de WebSphere Application Server para ejecutar el script Jython que acaba de crear.

Abra un indicador de mandatos y vaya al directorio `\bin` del directorio inicial de WebSphere Application Server, por ejemplo, directorio `C:\Program Files\IBM\WebSphere\AppServer1\bin` y, a continuación, escriba el siguiente mandato y pulse Intro:

```
wsadmin -lang jython -f convertWMQRA.py
```

Si se le solicita, escriba su nombre de usuario y contraseña.

Nota: Si ejecuta el script en un perfil que forma parte de una configuración de despliegue en red, el script actualiza todos los perfiles que deben actualizarse en dicha configuración. Puede que sea necesario realizar una resincronización completa si tiene incoherencias en el archivo de configuración preexistente.

5. Si está ejecutando en una configuración de despliegue de red, asegúrese de que los agentes de nodo estén completamente resincronizados. Para obtener más información, consulte Sincronización de nodos mediante la herramienta de scripts wsadmin o Adición, gestión y eliminación de nodos.
6. Detenga todos los servidores del perfil. Si el perfil forma parte de una configuración de despliegue de red, detenga también cualquier miembro de clúster de la configuración, detenga todos los agentes de nodo de la configuración y detenga el gestor de despliegue. Si el perfil contiene un agente administrativo, detenga el agente administrativo.
7. Ejecute el mandato **osgiCfgInit** desde el directorio *profile_root*/bin .

El mandato `osgiCfgInit` restablece la memoria caché de clase utilizada por el entorno de ejecución OSGi. Si el perfil forma parte de una configuración de despliegue de red, ejecute el mandato **`osgiCfgInit`** desde el directorio `profile_root/bin` de cada perfil que forme parte de la configuración.

8. Reinicie todos los servidores del perfil. Si el perfil forma parte de una configuración de despliegue de red, reinicie también cualquier miembro de clúster de la configuración, reinicie todos los agentes de nodo de la configuración y reinicie el gestor de despliegue. Si el perfil contiene un agente administrativo, reinicie el agente administrativo.
9. Repita el paso 2 para comprobar que el adaptador de recursos está ahora en el nivel correcto.

Qué hacer a continuación

Si sigue experimentando problemas después de realizar los pasos descritos en este tema, y ha utilizando previamente el botón **Actualizar adaptador de recursos** en el panel Valores de proveedor de JMS de la consola administrativa de WebSphere Application Server para actualizar el adaptador de recursos de IBM MQ en cualquier nodo del entorno, es posible que esté experimentando el problema descrito en el APAR PM10308.

Conceptos relacionados

Utilización del adaptador de recursos de IBM MQ

Información relacionada para WebSphere Application Server 8.5.5

Cómo asegurarse de que los servidores utilizan el último nivel de mantenimiento disponible de adaptador de recursos de IBM MQ

Sincronización de nodos mediante la herramienta de scripts wsadmin

Adición, gestión y eliminación de nodos

Valores del proveedor de JMS

Configurar la propiedad JMS PROVIDERVERSION

El proveedor de mensajería de IBM MQ tiene tres modalidades de operación: modalidad normal, modalidad normal con restricciones y modalidad de migración. Puede establecer la propiedad JMS **PROVIDERVERSION** para seleccionar cuál de estas modalidades utiliza una aplicación JMS para publicación y suscripción.

Acerca de esta tarea

La selección de la modalidad de operación del proveedor de mensajería IBM MQ se puede controlar principalmente estableciendo la propiedad de fábrica de conexiones PROVIDERVERSION. La modalidad de operación también se puede seleccionar automáticamente si no se ha especificado una modalidad.

La propiedad **PROVIDERVERSION** distingue entre las tres modalidades de operación de proveedor de mensajería de IBM MQ:

Modalidad normal del proveedor de mensajería IBM MQ
La modalidad normal utiliza todas las características de un gestor de colas IBM MQ para implementar JMS. Esta modalidad se optimiza para utiliza la API y las funciones de JMS 2.0.

Modalidad normal con restricciones del proveedor de mensajería IBM MQ
La modalidad normal con restricciones utiliza la API de JMS 2.0, pero no las nuevas características, es decir, suscripciones compartidas, entrega retardada y envío asíncrono.

Modalidad de migración del proveedor de mensajería IBM MQ
Con la modalidad de migración, puede conectarse a un gestor de colas de IBM MQ 8.0 o posterior, pero no se utiliza ninguna de las características de un gestor de colas de IBM WebSphere MQ 7.0 o posterior, como lectura anticipada o modalidad continua..

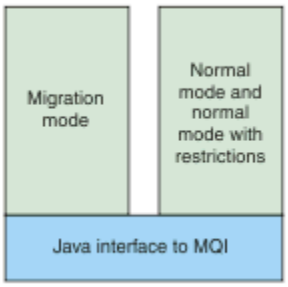


Figura 89. Modalidades de proveedor de mensajería

Procedimiento

Para configurar la propiedad **PROVIDERVERSION** para una fábrica de conexiones específica:

- Para configurar la propiedad **PROVIDERVERSION** utilizando IBM MQ Explorer, consulte Configurar gestores de colas y objetos.
- Para configurar la propiedad **PROVIDERVERSION** utilizando la herramienta de administración JMS, consulte Configurar gestores de colas y objetos.
- Para configurar la propiedad **PROVIDERVERSION** en una aplicación JMS utilizando las extensiones de IBM JMS o IBM MQ JMS, consulte Creación y configuración de fábricas de conexiones y destinos en una aplicación IBM MQ classes for JMS.

Para alterar temporalmente los valores de la modalidad del proveedor de fábrica de conexiones para todas las fábricas de conexiones de la JVM:

- Para alterar temporalmente los valores de modalidad de proveedor de fábrica de conexiones, utilice la propiedad `com.ibm.msg.client.wmq.overrideProviderVersion`

  Si no puede cambiar la fábrica de conexiones que está utilizando, puede utilizar la propiedad `com.ibm.msg.client.wmq.overrideProviderVersion` para alterar temporalmente cualquier valor de la fábrica de conexiones. Esta alteración temporal se aplica a todas las fábricas de conexiones de la JVM, pero los objetos de fábrica de conexiones reales no se modifican.

Conceptos relacionados

Resolución de problemas con la versión de proveedor JMS

Referencia relacionada

PROVIDERVERSION

Propiedades de fábrica de conexiones

Dependencias entre propiedades de objetos IBM MQ classes for JMS

Modalidades de operación del proveedor de mensajería de IBM MQ

Puede seleccionar qué modalidad de operación de proveedor de mensajería de IBM MQ utiliza una aplicación de JMS para publicar y suscribirse estableciendo la propiedad PROVIDERVERSION para la fábrica de conexiones en el valor adecuado. En algunos casos, la propiedad PROVIDERVERSION se establece como unspecified (sin especificar), en cuyo caso el cliente de JMS utiliza un algoritmo para determinar qué modalidad de operación se debe utilizar.

Valores de la propiedad PROVIDERVERSION

Puede establecer la propiedad **PROVIDERVERSION** de la fábrica de conexiones en cualquiera de los siguientes valores:

8 - modalidad normal

La aplicación JMS utiliza la modalidad normal. Esta modalidad utiliza todas las características de un gestor de colas de IBM MQ para implementar JMS.

7 - modalidad normal con restricciones
La aplicación JMS utiliza la modalidad normal con restricciones. Esta modalidad utiliza la API de JMS 2.0, pero no las nuevas características tales como suscripciones compartidas, entrega retrasada o envío asíncrono.

6- modalidad de migración
La aplicación JMS utiliza la modalidad de migración. En la modalidad de migración, IBM MQ classes for JMS utiliza características y algoritmos similares a los que se suministran con IBM WebSphere MQ 6.0.

unspecified (el valor predeterminado)
El cliente de JMS utiliza un algoritmo para determinar qué modalidad de operación se utiliza.

El valor que especifique para la propiedad **PROVIDERVERSION** debe ser una serie. Si va a especificar la opción 8, 7 o 6, puede hacerlo en cualquiera de los siguientes formatos:

- V.R.M.F
- V.R.M
- V.R
- V

donde V, R, M y F son valores enteros mayores que o iguales a cero. Los valores R, M y F adicionales son opcionales y están disponibles para su uso en caso de que se necesite un control de grano fino. Por ejemplo, si desea utilizar un nivel de **PROVIDERVERSION** de 7, puede establecer **PROVIDERVERSION**=7, `7.0`, `7.0.0` o `7.0.0.0`.

Tipos de objeto de fábrica de conexiones

Puede establecer la propiedad **PROVIDERVERSION** para los siguientes tipos de objeto de fábrica de conexiones:

- MQConnectionFactory
- MQQueueConnectionFactory
- MQTopicConnectionFactory
- MQXAConnectionFactory
- MQXAQueueConnectionFactory
- MQXAQueueConnectionFactory
- MQXAQueueConnectionFactory
- MQXATopicConnectionFactory

Para obtener más información sobre estos distintos tipos de fábrica de conexiones, consulte “Configuración de objetos JMS y Jakarta Messaging utilizando las herramientas de administración” en la página 721.

Conceptos relacionados
Proveedor de mensajería de IBM MQ

Modalidad normal de PROVIDERVERSION

La modalidad normal utiliza todas las características de un gestor de colas IBM MQ para implementar JMS. Esta modalidad se optimiza para utiliza la API y las funciones de JMS 2.0.

El siguiente diagrama de flujo muestra las comprobaciones que el cliente de JMS lleva a cabo para determinar si se puede crear una conexión de modalidad normal.

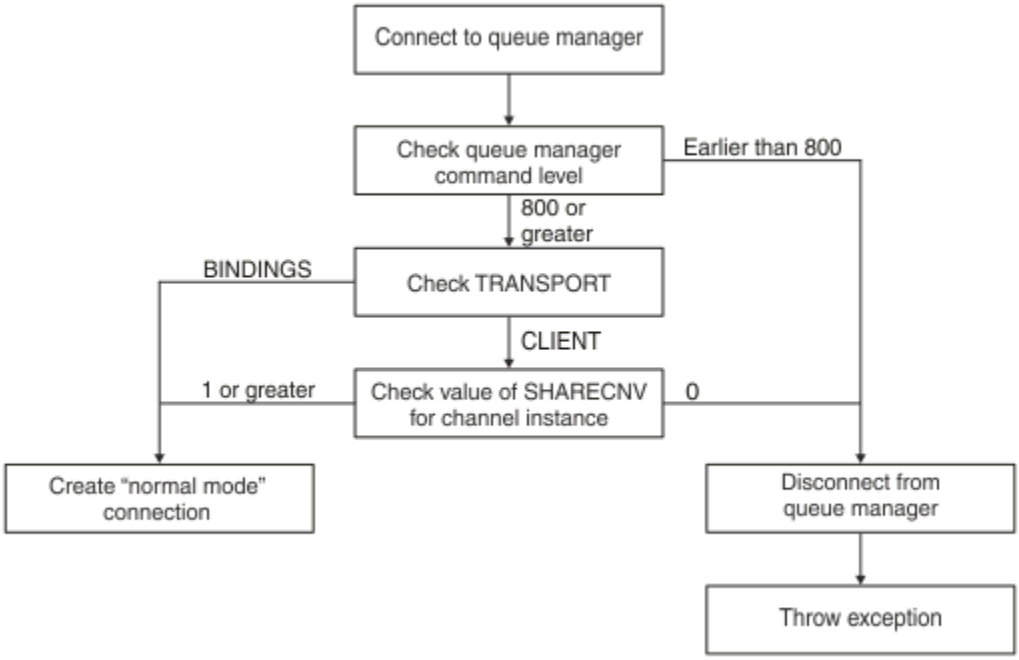


Figura 90. Modalidad normal de PROVIDERVERSION

Si el gestor de colas especificado en los valores de fábrica de conexiones tiene un nivel de mandato de 800 o superior, y la propiedad **TRANSPORT** de la fábrica de conexiones se establece en BINDINGS, se crea una conexión de modalidad normal sin comprobar más propiedades.

Si el gestor de colas especificado en los valores de la fábrica de conexiones tiene un nivel de mandato de 800 o superior, y la propiedad **TRANSPORT** se establece en CLIENT, también se comprueba la propiedad **SHARECNV** en el canal de conexión de servidor. Esta comprobación es necesaria porque la modalidad normal del proveedor de mensajería de IBM MQ utiliza la característica de compartimiento de conversaciones. Por lo tanto, para que un intento de conexión de modalidad normal sea satisfactorio, la propiedad **SHARECNV**, que controla el número de conversaciones que pueden compartirse, debe tener el valor 1 o superior.

Si todas las comprobaciones que se muestran en el diagrama de flujo son satisfactorias, se crea una conexión de modalidad normal al gestor de colas y se pueden utilizar todas las características y la API de JMS 2.0, es decir, envío asíncrono, entrega retardada y suscripción compartida.

Un intento de crear una conexión de modalidad normal falla por cualquiera de las siguientes razones:

- El gestor de colas especificado en los valores de fábrica de conexiones tiene un nivel de mandato que es anterior a 800. En este caso, el método `createConnection` falla con una excepción `JMSFMQ0003`.
- La propiedad **SHARECNV** del canal de conexión del servidor se establece en `0`. Si esta propiedad no tiene un valor de `1` o superior, el método `createConnection` falla con una excepción `JMSCC5007`.

Referencia relacionada

Dependencias entre propiedades de objetos IBM MQ classes for JMS
DEFINE CHANNEL (propiedad SHARECNV)
TRANSPORT

Modalidad normal con restricciones de PROVIDERVERSION

La modalidad normal con restricciones utiliza la API de JMS 2.0, pero no las nuevas características de IBM MQ 8.0 o posterior tales como las suscripciones compartidas, el entrega retardada o el envío asíncrono.

El siguiente diagrama de flujo muestra las comprobaciones que el cliente JMS realiza para determinar si se puede crear una conexión de modalidad normal con restricciones.

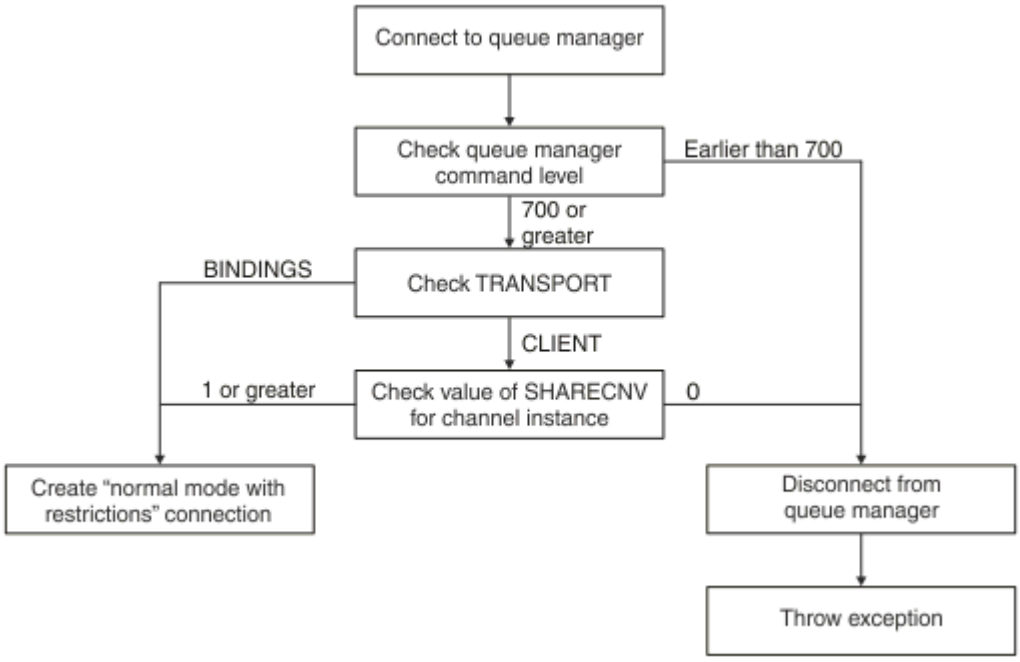


Figura 91. Modalidad normal con restricciones de PROVIDERVERSION

Si el gestor de colas especificado en los valores de fábrica de conexiones tiene un nivel de mandato de 700 o superior, y la propiedad **TRANSPORT** de la fábrica de conexiones se establece en BINDINGS, se crea una conexión de modalidad normal sin comprobar más propiedades.

Si el gestor de colas especificado en los valores de la fábrica de conexiones tiene un nivel de mandato de 700 o superior, y la propiedad **TRANSPORT** se establece en CLIENT, también se comprueba la propiedad **SHARECNV** en el canal de conexión de servidor. Esta comprobación es necesaria porque la modalidad normal con restricciones del proveedor de mensajería de IBM MQ utiliza la característica de compartimiento de conversaciones. Por lo tanto, para que un intento de conexión de modalidad normal con restricciones sea satisfactorio, la propiedad **SHARECNV**, que controla el número de conversaciones que pueden compartirse, debe tener el valor 1 o superior.

Si todas las comprobaciones que se muestran en el diagrama de flujo son satisfactorias, se crea una conexión de modalidad normal con restricciones al gestor de colas y podrá utilizar la API de JMS 2.0, pero no las características de envío asíncrono, entrega retardada o suscripción compartida.

Un intento de crear una conexión de modalidad normal con restricciones falla por cualquiera de las siguientes razones:

- El gestor de colas especificado en los valores de fábrica de conexiones tiene un nivel de mandato que es anterior a 700. En este caso, el método `createConnection` falla con la excepción `JMSFCC5008`.
- La propiedad **SHARECNV** del canal de conexión del servidor se establece en `0`. Si esta propiedad no tiene un valor de 1 o superior, el método `createConnection` falla con una excepción `JMSCC5007`.

Referencia relacionada

Dependencias entre propiedades de objetos IBM MQ classes for JMS
DEFINE CHANNEL (propiedad SHARECNV)
TRANSPORT

Modalidad de migración de PROVIDERVERSION

Para la modalidad de migración, IBM MQ classes for JMS utiliza características y algoritmos similares a los que se suministran con IBM WebSphere MQ 6.0, como por ejemplo, publicación/suscripción, selección implementada en el lado del cliente, canales no multiplex y el sondeo utilizado para implementar escuchas.

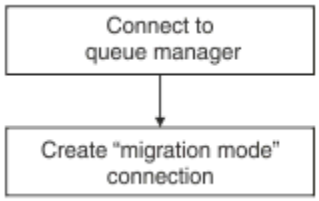


Figura 92. Modalidad de migración de PROVIDERVERSION

Si desea conectar a WebSphere Message Broker 6.0 o WebSphere Message Broker 6.1 utilizando IBM MQ Enterprise Transport versión 6.0, debe utilizar la modalidad de migración.

Puede conectarse a un gestor de colas de IBM MQ 8.0 utilizando la modalidad de migración, pero no se utiliza ninguna de las nuevas características de un gestor de colas de IBM MQ classes for JMS, como por ejemplo, la lectura anticipada o la modalidad continua. Si tiene un cliente IBM MQ 8.0 o un cliente posterior que se conecta a un gestor de colas IBM MQ 8.0 o posterior en una plataforma distribuida, z/OS o un gestor de colas IBM MQ for z/OS 8.0 o posterior, la selección de mensajes la realiza el gestor de colas en lugar de en el sistema cliente.

Si se especifica la modalidad de migración del proveedor de mensajería de IBM MQ y IBM MQ classes for JMS intenta utilizar cualquiera de la API de JMS 2.0, la llamada al método de API falla con la excepción `JMSCC5007`.

Referencia relacionada

Dependencias entre propiedades de objetos IBM MQ classes for JMS

TRANSPORT

PROVIDERVERSION sin especificar

Cuando la propiedad **PROVIDERVERSION** de una fábrica de conexiones tiene el valor unspecified (sin especificar), el cliente de JMS utiliza un algoritmo para determinar qué modalidad de operación se utiliza para conectarse al gestor de colas. Una fábrica de conexiones que se creó en el espacio de nombres JNDI con una versión anterior de IBM MQ classes for JMS adopta el valor unspecified (sin especificar) cuando la fábrica de conexiones se utiliza con la nueva versión de IBM MQ classes for JMS.

Si la propiedad **PROVIDERVERSION** tiene el valor unspecified (sin especificar), el algoritmo se utiliza cuando se llama al método `createConnection`. El algoritmo comprueba diversas propiedades de fábrica de conexiones para determinar si se requiere la modalidad normal de proveedor de mensajería, la modalidad normal con restricciones de IBM MQ o la modalidad de migración de proveedor de mensajería de IBM MQ. Siempre se intenta primero la modalidad normal y después la modalidad normal con restricciones. Si no puede realizarse ninguno de estos tipos de conexión, el cliente de JMS se desconecta del gestor de colas y, a continuación, se vuelve a conectar al gestor de colas para intentar una conexión en modalidad de migración.

Comprobación de las propiedades BROKERVER, BROKERQMGR, PSMODE y BROKERCONQ

La comprobación de valores de propiedades empieza con la propiedad **BROKERVER** tal como se muestra en la Figura 1.

Si la propiedad **BROKERVER** se establece en V1, la propiedad **TRANSPORT** se comprueba a continuación, tal como se muestra en la Figura 2. Sin embargo, si la propiedad **BROKERVER** se establece en V2, la comprobación adicional que se muestra en la Figura 1 se realiza antes de que se compruebe la propiedad **TRANSPORT**.

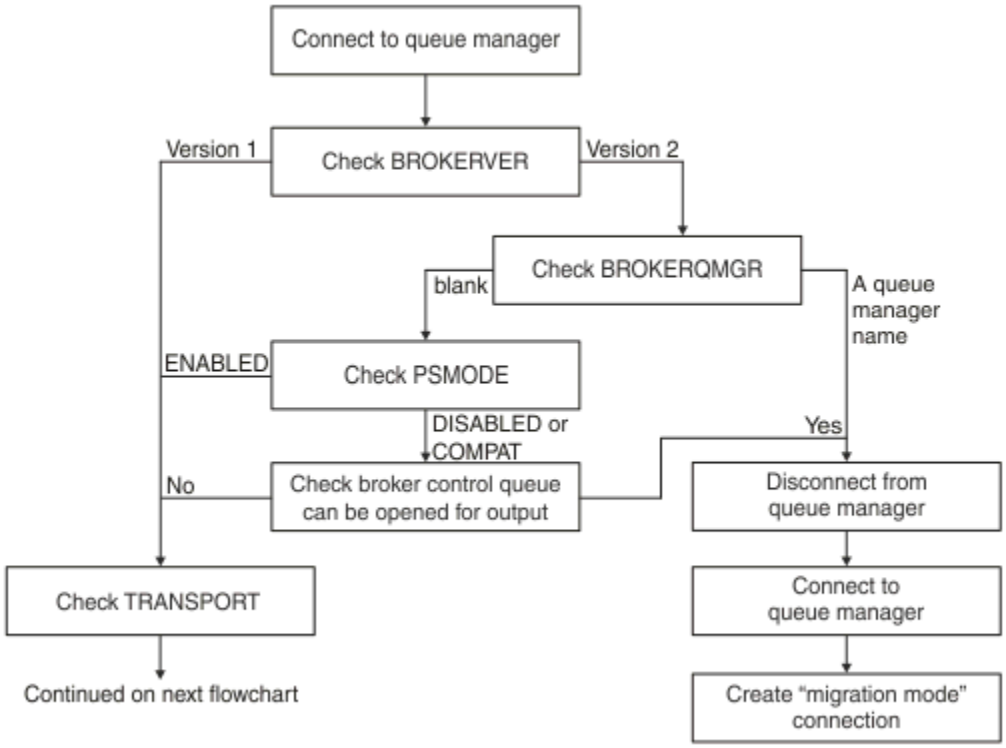


Figura 93. PROVIDERVERSION sin especificar

Si la propiedad **BROKERVER** se establece en V2, para que una conexión en modalidad normal sea posible, la propiedad **BROKERQMGR** debe ser en blanco `blank`. Además, el atributo **PSMODE** del gestor de colas debe establecerse en `ENABLED` o la cola de control de intermediario especificada por la propiedad **BROKERCONQ** no debe poder abrirse para la salida.

Si los valores de propiedad se establecen como necesarios para una conexión de modalidad normal, la comprobación siguiente pasa a la propiedad **TRANSPORT** tal como se muestra en la Figura 2.

Si los valores de propiedad no se han establecido como necesarios para una conexión en modalidad normal, el cliente de JMS se desconecta del gestor de colas y, a continuación, se vuelve a conectar y crea una conexión en modalidad de migración. Esto sucede en los siguientes casos:

- Si la propiedad **BROKERQMGR** es `blank` y el atributo **PSMODE** del gestor de colas se establece en `COMPAT` o `DISABLED` y la cola de control de intermediario especificada por la propiedad **BROKERCONQ** puede abrirse para la salida (es decir, `MQOPEN` para la salida se ejecuta correctamente).
- Si la propiedad **BROKERQMGR** especifica un nombre de cola.

Comprobación de la propiedad TRANSPORT y el nivel de mandato

En la Figura 2 se muestran las comprobaciones que se llevan a cabo para la propiedad **TRANSPORT** y el nivel de mandato del gestor de colas.

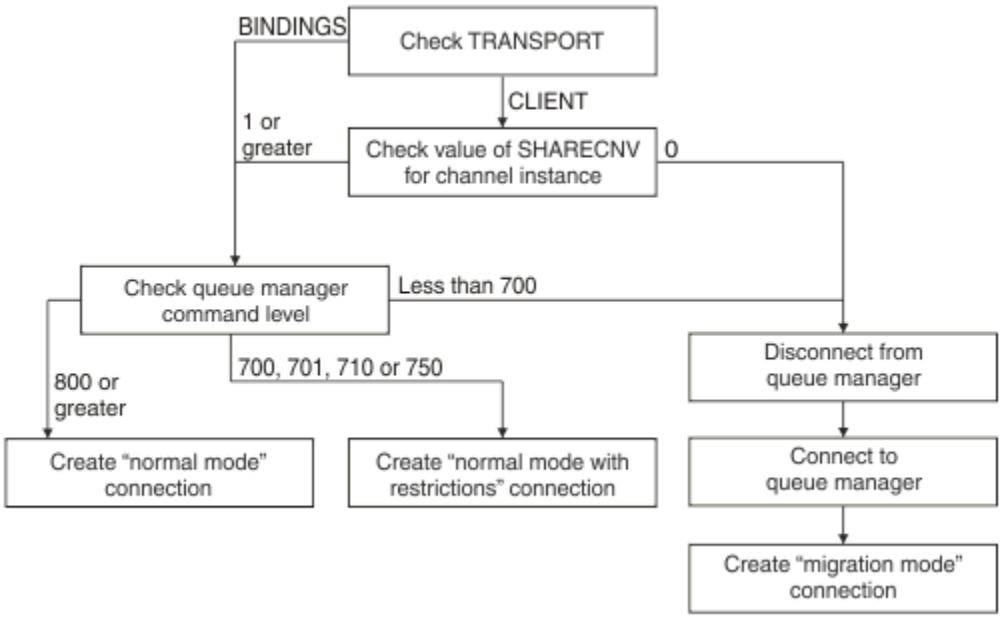


Figura 94. PROVIDERVERSION sin especificar (continuación)

Una conexión en modalidad normal se crea en cualquiera de los casos siguientes:

- La propiedad **TRANSPORT** de la fábrica de conexiones se establece en BINDINGS y el gestor de colas tiene un nivel de mandato de 800 o superior.
- La propiedad **TRANSPORT** se establece en CLIENT, la propiedad **SHARECNV** del canal de conexión de servidor tiene un valor de 1 o superior, y el gestor de colas tiene un nivel de mandato de 800 o superior.

Si el gestor de colas tiene un nivel de mandatos de 750, se crea una modalidad normal con restricciones de conexión con el gestor de colas.

Una conexión en modalidad de migración también se crea si la propiedad **TRANSPORT** se establece en CLIENT y la propiedad **SHARECNV** en el canal de conexión de servidor tiene un valor de 0.

Referencia relacionada

Dependencias entre propiedades de objetos IBM MQ classes for JMS
ALTER QMGR (atributo PSMODE)
BROKERCONQ
BROKERQMGR
BROKERVER
DEFINE CHANNEL (propiedad SHARECNV)
TRANSPORT

Configurar información de versión de proveedor en WebSphere Application Server

Para configurar información de versión de proveedor en WebSphere Application Server, puede utilizar la consola administrativa o mandatos wsadmin.

Procedimiento

Para configurar información de versión de proveedor para un objeto de fábrica de conexiones o especificación de activación de IBM MQ en WebSphere Application Server, consulte la *Información relacionada* para ver los enlaces a información adicional de la documentación del producto WebSphere Application Server.

Información relacionada para WebSphere Application Server 8.5.5
Valores de fábrica de conexiones del proveedor de mensajería IBM MQ
Mandato **createWMQConnectionFactory**
Valores de especificación de activación del proveedor de mensajería IBM MQ
Mandato **createWMQActivationSpec**
Información relacionada para WebSphere Application Server 8.0.0
Valores de fábrica de conexiones del proveedor de mensajería IBM MQ
Mandato **createWMQConnectionFactory**
Valores de especificación de activación del proveedor de mensajería IBM MQ
Mandato **createWMQActivationSpec**
Información relacionada para WebSphere Application Server 7.0.0
Valores de fábrica de conexiones del proveedor de mensajería IBM MQ
Mandato **createWMQConnectionFactory**
Valores de especificación de activación del proveedor de mensajería IBM MQ
Mandato **createWMQActivationSpec**

Eliminación de suscripciones duraderas de WebSphere Application Server

Cuando se utiliza el proveedor de mensajería IBM MQ con WebSphere Application Server 7.0 y WebSphere Application Server 8.0, las suscripciones duraderas creadas por aplicaciones de bean controlado por mensaje enlazadas a especificaciones de activación no se eliminan. Las suscripciones duraderas se pueden eliminar utilizando IBM MQ Explorer o un programa de utilidad de línea de mandatos IBM MQ.

Acerca de esta tarea

Una aplicación de bean controlado por mensaje que elimina una suscripción duradera se puede configurar para utilizar un puerto de escucha o bien una especificación de activación, siempre que la aplicación se esté ejecutando dentro de una instancia de WebSphere Application Server 7.0 o WebSphere Application Server 8.0 que utilice la modalidad normal del proveedor de mensajería de IBM MQ para conectarse a IBM MQ.

Si la aplicación de bean controlado por mensaje está enlazada a un puerto de escucha, el proveedor de mensajería IBM MQ crea la suscripción duradera para la aplicación la primera vez que se inicia la aplicación. La suscripción duradera se elimina cuando la aplicación de bean controlado por mensaje se desinstala de un servidor de aplicaciones y el servidor de aplicaciones se reinicia.

Una aplicación de bean controlado por mensaje que está enlazada a una especificación de activación funciona de una forma ligeramente distinta. Se crea la suscripción duradera para la aplicación, la primera vez que se inicia la aplicación. Sin embargo, la suscripción duradera no se elimina cuando la aplicación se desinstala y el servidor de aplicaciones se reinicia.

Esto puede dar a lugar a que una serie de suscripciones duraderas permanezca en un motor de publicación/suscripción de IBM MQ para las aplicaciones que ya no están instaladas en un sistema WebSphere Application Server. Estas suscripciones se conocen como "suscripciones huérfanas" y pueden provocar problemas en el gestor de colas cuando se ejecuta el motor de publicación/suscripción.

Cuando se publica un mensaje en un tema, el motor de publicación/suscripción de IBM MQ hace una copia de ese mensaje para cada suscripción duradera que está registrada en ese tema y se coloca en una cola interna. Las aplicaciones que utilizan esa suscripción duradera seleccionarán y consumirán el mensaje de esta cola interna.

Si la aplicación de bean controlado por mensaje que estaba utilizando esa suscripción duradera deja de estar instalada, las copias de los mensajes publicados para la aplicación se seguirán realizando. Sin embargo, estos mensajes nunca se procesarán, lo que significa que podría haber un gran número de mensajes restantes en la cola interna que nunca se eliminarán.

Antes de empezar

Las suscripciones que están registradas con el motor de publicación/suscripción IBM MQ tendrán un nombre de suscripción asociado.

Las suscripciones duraderas creadas por el proveedor de mensajería WebSphere Application Server IBM MQ para beans controlados por mensajes que están enlazados a especificaciones de activación tendrán un nombre de suscripción con el formato siguiente.

```
JMS:queue manager name:client identifier:subscription name
```

Donde:

nombre de gestor de colas
Este es el nombre del gestor de colas IBM MQ donde se está ejecutando el motor de publicación/suscripción.

identificador cliente
Este es el valor de la propiedad de ID de cliente de la especificación de activación a la que está enlazado el bean controlado por mensaje.

nombre suscripción
Este es el valor de la propiedad Nombre de suscripción de la especificación de activación para cuyo uso se ha configurado el bean controlado por mensaje.

Por ejemplo, supongamos que se tiene una especificación de activación que se ha configurado para conectarse al gestor de colas testQM. La especificación de activación tiene las propiedades siguientes establecidas:

- ID de cliente = testClientID
- Nombre de suscripción = durableSubscription1

Si un bean controlado por mensaje extrae una suscripción duradera que está enlazada a esta especificación de activación, se crea una suscripción en el motor de publicación/suscripción IBM MQ en el gestor de colas testQM que tiene el nombre de suscripción siguiente:

- JMS:testQM:testClientID:durableSubscription1

Las suscripciones que se han registrado con el motor de publicación/suscripción IBM MQ para un gestor de colas especificado se pueden visualizar de una de las dos formas siguientes:

- La primera opción es utilizar el MQ Explorer. Cuando MQ Explorer se ha conectado a un gestor de colas que se está utilizando para el trabajo de publicación/suscripción, la lista de suscriptores que están registrados actualmente con el motor de publicación/suscripción se puede ver pulsando la entrada `IBM WebSphere MQ ->queue manager name-> Subscriptions` en el panel de navegación.
- La otra forma para ver las suscripciones que se han registrado con un motor de publicación/suscripción es utilizar el programa de utilidad de línea de mandatos de IBM MQ **runmqsc** y ejecutar el mandato **display sub**. Para ello, inicie un indicador de mandatos, vaya al directorio *WebSphere MQ*\bin y especifique el mandato siguiente para iniciar **runmqsc**:

  – `runmqsc queue manager name`

  Cuando se ha iniciado el programa de utilidad **runmqsc**, especifique el mandato siguiente para lista todas las suscripciones duraderas registradas actualmente con el motor de publicación/suscripción que se ejecuta en el gestor de colas al que se ha conectado **runmqsc**.

  – `display sub(*) durable`

Para comprobar si las suscripciones duraderas registradas con los motores de publicación/suscripción siguen activas:

1. Genere la lista de suscripciones duraderas que se han registrado con el motor de publicación/suscripción.
2. Para cada suscripción duradera:

- Consulte el nombre de la suscripción para el suscriptor duradero,, y anote el valor de *identificador cliente* y *nombre suscripción*.
- Consulte los sistemas WebSphere Application Server que se conectan a este motor de publicación/suscripción. Consulte si hay alguna especificación de activación definida que tenga la propiedad ID de cliente que coincida con el valor *identificador cliente* y la propiedad de nombre de suscripción que coincida con el *nombre suscripción*.
- Si no se encuentra ninguna especificación de activación que tenga las propiedades ID de cliente y Nombre de suscripción que coincidan con los campos *identificador cliente* y *nombre suscripción* en el nombre de suscripción de IBM MQ, no hay ninguna especificación de activación que utilice esta suscripción duradera. La suscripción duradera se puede suprimir.
- Si hay una especificación de activación definida que coincida con el nombre de suscripción duradera, se debe realizar la comprobación final para ver si hay una aplicación de bean controlado por mensaje que utiliza esta especificación de activación. Para ello:
  - Anote el nombre JNDI para la especificación de activación que ha extraído la suscripción duradera que está consultando actualmente.
  - Abra el panel de configuración en la consola de administración de WebSphere Application Server para cada aplicación de bean controlado por mensaje que se ha instalado.
  - Pulse el enlace de enlaces de escucha del bean controlado por mensaje en el panel de configuración.
  - Se muestra una tabla con información sobre la aplicación de bean controlado por mensaje. Si el botón de opción de especificación de activación está seleccionado en la columna Enlaces y el campo del nombre JNDI de recurso de destino contiene el nombre JNDI para la especificación de activación que ha extraído la suscripción duradera, la suscripción se sigue utilizando y no se puede suprimir.
  - Si no se puede encontrar ninguna aplicación de bean controlado por mensaje que esté utilizando la especificación de activación, la suscripción duradera se puede suprimir.

Procedimiento

Una vez que se ha identificado una suscripción duradera "huérfana", se puede suprimir utilizando IBM MQ Explorer o el IBM MQ programa de utilidad de línea de mandatos **runmqsc**.

Para suprimir una suscripción duradera "huérfana" utilizando IBM MQ Explorer:

1. Resalte la entrada para la suscripción
2. Pulse con el botón derecho del ratón en la entrada y seleccione **Suprimir ...** en el menú. Se visualizará una ventana de confirmación.
3. Compruebe que el nombre de suscripción que se muestra en la ventana de confirmación es correcto y pulse **Sí**.

Ahora IBM MQ Explorer suprime la suscripción del motor de publicación/suscripción y borra los recursos internos asociados (como mensajes no procesados que se han publicado para el tema en el estaba registrada la suscripción duradera).

Para suprimir una suscripción duradera "huérfana" utilizando el IBM MQ programa de utilidad de línea de mandatos **runmqsc**, se debe ejecutar el mandato **delete sub** :

1. Abra una sesión de indicador de mandatos
2. Vaya al directorio *IBM MQ*\bin
3. Especifique el mandato siguiente para iniciar **runmqsc**:

```
runmqsc queue manager name
```

4. Cuando se ha iniciado el programa de utilidades **runmqsc**, especifique:

```
delete sub(Subscription name)
```

donde *nombre suscripción* es el nombre de suscripción de la suscripción duradera que tiene el formato:

- JMS:*queue manager name*:*client identifier*:*subscription name*

Configuración de Managed File Transfer

Puede configurar las características de Managed File Transfer después de la instalación.

Puede aprovechar las soluciones de alta disponibilidad de IBM MQ para mejorar la resiliencia de la configuración de Managed File Transfer . Si los agentes utilizan gestores de colas de datos replicados (RDQM), debe configurarlos para utilizar la característica de dirección IP flotante. Esto significa que los agentes utilizan la misma dirección IP para comunicarse con cualquiera de las tres instancias RDQM que se están ejecutando actualmente y se reconectan automáticamente en la migración tras error (consulte Alta disponibilidad RDQM y Creación y supresión de una dirección IP flotante). Si utiliza la solución de gestor de colas de varias instancias, las aplicaciones utilizan una dirección IP diferente para comunicarse con cada instancia, que es manejada por la reconexión del cliente en la migración tras error (consulte “Gestores de colas multiinstancia” en la página 530 y “Reconexión de canal y cliente” en la página 587).

Conceptos relacionados

Consejos y sugerencias para utilizar Managed File Transfer

Tareas relacionadas

Supervisión de recursos de MFT

Personalización de MFT con salidas de usuario

Configuración de MQMFTCredentials.xml

Protección de Managed File Transfer

Especificación de programas que se van a ejecutarse con MFT

Resolución de problemas de Managed File Transfer

Administración de Managed File Transfer

Referencia relacionada

Mandatos de MFT

El archivo MFT `agent.properties`

Recuperación y reinicio de MFT

Opciones de configuración de MFT en Multiplatforms

Managed File Transfer proporciona un conjunto de archivos de propiedades que contienen información clave sobre la configuración y son necesarios para la operación. Estos archivos de propiedades están en el directorio de configuración que ha definido al instalar el producto.

Puede tener varios conjuntos de opciones de configuración, cada conjunto de opciones de configuración contiene un conjunto de directorios y archivos de propiedades. Los valores definidos en estos archivos de propiedades se utilizan como los parámetros predeterminados para todos los mandatos de Managed File Transfer, a menos que se especifique explícitamente un valor distinto en la línea de mandatos.

Para cambiar el conjunto predeterminado de opciones de configuración que está utilizando, puede utilizar el mandato **fteChangeDefaultConfigurationOptions**. Para cambiar el conjunto de opciones de configuración que está utilizando para un mandato individual, puede utilizar el parámetro **-p** con cualquier mandato de Managed File Transfer.

El nombre de un conjunto de opciones de configuración es el nombre del gestor de colas de coordinación y se recomienda no cambiarlo. Sin embargo, es posible cambiar el nombre de un conjunto de opciones de configuración, pero debe cambiar el nombre de los directorios `config` y `logs`. En

los ejemplos siguientes, el nombre del conjunto de opciones de configuración se representa como *nombre_gestcolas_coordinación*.

Estructura de directorios de las opciones de configuración

Cuando configura el producto, se crean archivos de directorios y propiedades en la estructura siguiente del directorio de configuración. También puede cambiar estos directorios y archivos de propiedades con los mandatos siguientes: **fteSetupCoordination**, **fteSetupCommands**, **fteChangeDefaultConfiguration** y **fteCreateAgent**.

```
MQ_DATA_PATH/mqft/
    config/
       coordination_qmgr_name/
            coordination.properties
            command.properties
            agents/
               agent_name/
                    agent.properties
                    exits
            loggers/
               logger_name
                    logger.properties
    installations/
       installation_name/
            installation.properties
```

El directorio *nombre_gestcolas_coordinación* es un directorio de opciones de configuración. Hay más de un directorio de opciones de configuración en el directorio de configuración. El directorio *nombre_agente* es un directorio de agentes. Además de contener el archivo `agent.properties`, este directorio contiene el directorio `exits`, que es la ubicación predeterminada para rutinas de salida de usuario y diversos archivos XML generados por los mandatos **fteCreateBridgeAgent** y **fteCreateCDAgent**. Puede haber más de un directorio de agente en el directorio `agents` de un conjunto de opciones de configuración.

Archivos de propiedades

installation.properties
: El archivo `installation.properties` especifica el nombre del conjunto predeterminado de opciones de configuración. Esta entrada apunta Managed File Transfer a un conjunto estructurado de directorios y archivos de propiedades que contienen la configuración que se va a utilizar. Normalmente, el nombre de un conjunto de opciones de configuración es el nombre del gestor de colas de coordinación asociado. Para obtener más información sobre el archivo `installation.properties`, consulte El archivo installation.properties MFT.

coordination.properties
: El archivo `coordination.properties` especifica los detalles de conexión al gestor de colas de coordinación. Dado que varias instalaciones de Managed File Transfer pueden compartir el mismo gestor de colas de coordinación, puede utilizar un enlace simbólico a un archivo `coordination.properties` común en una unidad compartida. Para obtener más información sobre el archivo `coordination.properties`, consulte El archivo installation.properties MFT.

command.properties
: El archivo command.properties de MFT especifica el gestor de colas de mandatos al que se conectarse cuando se emiten mandatos y la información que requiere Managed File Transfer para contactar con ese gestor de colas. Para obtener más información sobre el archivo `command.properties`, consulte El archivo installation.properties MFT.

agent.properties
: Cada Managed File Transfer Agent tiene su propio archivo de propiedades, `agent.properties`, que debe contener la información que un agente utiliza para conectarse a su gestor de colas. El archivo `agent.properties` también puede contener propiedades que alteran el comportamiento del agente. Para obtener más información sobre el archivo `agent.properties`, consulte El archivo MFT `agent.properties`.

logger.properties
El archivo `logger.properties` especifica las propiedades de configuración para los registradores. Para obtener más información sobre el archivo `logger.properties`, consulte Propiedades de configuración del registrador MFT.

Archivos de propiedades y páginas de códigos

El contenido de todos los archivos de propiedades de Managed File Transfer debe estar en inglés de Estados Unidos debido a una limitación de Java. Si edita los archivos de propiedades en un sistema que no está en inglés de Estados Unidos, deberá utilizar secuencias de escape Unicode.

Referencia relacionada

Propiedades SSL/TLS para MFT

Propiedades del sistema Java para MFT

fteChangeDefaultConfigurationOptions

fteSetupCommands: crear el archivo command.properties de MFT

fteSetupCoordination

fteCreateAgent

z/OS MFT configuration options on z/OS

The Managed File Transfer configuration options on z/OS are the same as the options for distributed platforms.

For more information about configuration options on Multiplatforms, see “Opciones de configuración de MFT en Multiplatforms” on page 755.

On z/OS, the configuration location is defined by the environment variable BFG_DATA. If a configuration does not already exist under the z/OS UNIX System Services directory that is referenced by BFG_DATA, the BFGCUSTM JCL script of an MFT command PDSE library data set generates the jobs required to create the configuration. The configuration is then created when you run these generated jobs. Configuration creation relies on BFG_DATA referencing an existing directory that is accessible.

You can also create and maintain a configuration by using the same **fte** commands that are available on both Multiplatforms and z/OS. For a list of the **fte** commands, see MFT commands .

Related concepts

“Opciones de configuración de MFT en Multiplatforms” on page 755
Managed File Transfer proporciona un conjunto de archivos de propiedades que contienen información clave sobre la configuración y son necesarios para la operación. Estos archivos de propiedades están en el directorio de configuración que ha definido al instalar el producto.

“Creating an agent” on page 774
You need to copy the PDSE to make the agent-specific PDSE, for example *user.MFT.AGENT1*. Copy the PDSE from a previous agent or logger configuration, if they exist. If this is your first configuration, copy the PDSE supplied with MFT.

“Defining the coordination queue manager” on page 772
Managed File Transfer requires a queue manager to be created that acts as the coordination queue manager.

Related tasks

z/OS Configuring `MQMFTCredentials.xml` on z/OS

“Updating an existing MFT Agent or Logger command data set on z/OS” on page 775
You can update an Managed File Transfer command PDSE library data set that is created from the Managed File Transfer command template data set.

Windows Linux Descarga y configuración de Redistributable Managed File Transfer components

Redistributable Managed File Transfer package proporciona Redistributable Managed File Transfer Agent, que puede configurar para conectarse a una infraestructura de IBM MQ existente y permitir a los usuarios transferir los archivos sin tener que instalar IBM MQ. A partir de IBM MQ 9.3.0, el paquete redistribuible también incluye Redistributable Managed File Transfer Logger.

Antes de empezar

Para obtener información sobre los términos de licencia redistribuible para Redistributable Managed File Transfer Agent y Redistributable Managed File Transfer Logger, consulte Componentes redistribuibles deIBM MQ.

Los componentes de Redistributable Managed File Transfer package proporcionan la funcionalidad de Managed File Transfer con estas excepciones:

- Para Redistributable Managed File Transfer Agent, no se da soporte a la conexión de modalidad de enlaces a los gestores de colas de coordinación, mandatos y agente, debe utilizar la conexión de modalidad de cliente. Al ejecutar mandatos, hay que proporcionar los parámetros que son opcionales cuando se usa el Managed File Transfer que se instala como parte de IBM MQ: nombre, puerto y host del gestor de colas, y nombre del canal.
- Redistributable Managed File Transfer Logger sólo da soporte a registradores de tipo FILE, que se conectan en modalidad de cliente sólo al gestor de colas de coordinación. La conexión de modalidad de cliente al gestor de colas de coordinación para un registrador de base de datos no está soportada. Si requiere una conexión de modalidad de enlaces, debe utilizar una instalación estándar de IBM MQ.
- A partir de IBM MQ 9.3.0, el mandato **`fteCreateCDAgent.cmd`** no se incluye. Si desea una lista completa de mandatos disponibles, consulte Conjuntos de mandatos MFT instalados.
- Managed File Transfer Connect:Direct no recibe soporte.
- IBM MQ Explorer no se incluye.

Windows Debe instalar las bibliotecas de Microsoft Visual C++ Redistributable for Visual Studio 2015, 2017 and 2019, disponibles desde Microsoft, en el sistema para utilizar Redistributable Managed File Transfer Agent. Consulte Descargas más recientes compatibles de Visual C++.

A partir de IBM MQ 9.3.0, las bibliotecas Microsoft Visual C++ Redistributable for Visual Studio 2015, 2017 and 2019 también son necesarias para Redistributable Managed File Transfer Logger.

Nota: Advanced Message Security no está soportado con Redistributable Managed File Transfer package.

Acerca de esta tarea

De forma opcional, puede descargar Redistributable Managed File Transfer package y configurar Redistributable Managed File Transfer Agent para conectarse a una infraestructura de IBM MQ para permitir que los usuarios transfieran archivos entre su entorno local y la infraestructura IBM MQ existente sin que tengan que instalar IBM MQ para obtener la funcionalidad de Managed File Transfer.

A partir de IBM MQ 9.3.0, Redistributable Managed File Transfer package incluye también Redistributable Managed File Transfer Logger, que le permite configurar un registrador de archivos para conectarse en modalidad de cliente al gestor de colas de coordinación.

Procedimiento

1. Descargue el paquete IBM MQ redistribuible Managed File Transfer Agent de Fix Central.
   a) Elija el paquete correspondiente a su sistema operativo.

      Los nombres de archivado o de archivo .zip describen el contenido del archivo y los niveles de mantenimiento equivalentes. Los nombres de archivo tienen el formato siguiente:

- Windows *V.R.M.F*-IBM-MQFA-Redist-Win64
- Linux *V.R.M.F*-IBM-MQFA-Redist-LinuxX64
- Linux *V.R.M.F*-IBM-MQFA-Redist-LinuxS390X
- Linux *V.R.M.F*-IBM-MQFA-Redist-LinuxPPC64LE

donde *V.R.M.F* es el número de versión, por ejemplo, 9.2.0.0 o 9.2.1.0.

b) Identifique el directorio donde desee extraer el paquete, por ejemplo:

- Windows C:\MFTZ
- Linux /home/MFTZ

2. Extraiga el contenido del paquete descargado:

- Windows En Windows, utilice las herramientas de Windows Explorer para la extracción.
- Linux En Linux, extraiga y descomprima como se indica a continuación:

```
gunzip V.R.M.F-IBM-MQFA-Redist-LinuxX64.tar.gz
```

y, a continuación,

```
tar xvf V.R.M.F-IBM-MQFA-Redist-LinuxX64.tar
```

donde *V.R.M.F* es el número de versión, por ejemplo, 9.3.0.0 o 9.3.1.0.

Se crearán los directorios siguientes:

- Windows Linux bin: Contiene todos los mandatos necesarios de MFT
- Windows bin64: Contiene las bibliotecas necesarias que se precisan para el soporte de SO Windows de 64 bits
- Windows Linux java: Contiene las bibliotecas IBM JRE y IBM MQ
- Windows Linux licenses: Contiene los archivos de licencia
- Windows META-INF: Contiene archivos que tienen información de firma de código
- Windows Linux mqft: contiene los directorios ant y lib que son necesarios para el soporte de Ant y para el soporte de la función principal de MFT
- Windows Linux swtag: Contiene el archivo swidtag que necesitan los gestores de licencias para identificar las instalaciones en la máquina

Qué hacer a continuación

Está preparado para configurar un Managed File Transfer Agent. Para ver los pasos siguientes, consulte “Creación de la configuración inicial para Redistributable Managed File Transfer Agent” en la página 760.

A partir de IBM MQ 9.3.0, también puede configurar un Managed File Transfer Logger. Para ver los pasos siguientes para configurar el registrador, consulte “Creación de la configuración inicial para Redistributable Managed File Transfer Logger” en la página 762.

Referencia relacionada

Posibles errores al configurar Redistributable Managed File Transfer components

Windows Linux Creación de la configuración inicial para Redistributable Managed File Transfer Agent

Se puede configurar un Managed File Transfer Agent para conectar con una configuración existente de IBM MQ.

Antes de empezar

Asegúrese de haber descargado y extraído el contenido del paquete del Redistributable Managed File Transfer Agent. Para obtener más información, consulte “Descarga y configuración de Redistributable Managed File Transfer components” en la página 758.

Acerca de esta tarea

En primer lugar, cree el entorno que necesita Redistributable Managed File Transfer Agent. A continuación, puede configurar la conectividad con el gestor de colas que se ejecuta en el servidor de IBM MQ y, a continuación, configurar un agente y el gestor de colas de agente, antes de iniciar y verificar el agente.

A partir de IBM MQ 9.3.0, el entorno que crea se comparte con Redistributable Managed File Transfer Logger. Para obtener más información, consulte “Creación de la configuración inicial para Redistributable Managed File Transfer Logger” en la página 762.

Procedimiento

1. Cree el entorno del Redistributable Managed File Transfer Agent.

   Cuando ejecuta Mandato **fteCreateEnvironment**, se crea el directorio de datos MFT con la información de configuración para los agentes de MFT . Asegúrese de que está en el directorio `bin` que se ha creado al extraer el componente Redistributable Managed File Transfer Agent descargado. Ejecute el siguiente mandato:

   - Windows

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

   - Linux

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

   Este mandato tiene los siguientes parámetros opcionales:

   -d
   Este parámetro especifica la ubicación de la vía de acceso a datos en la que se crea, almacena y mantiene la configuración de MFT. Si ejecuta **fteCreateEnvironment** sin especificar la ubicación de datos, se crea el directorio `mftdata` en la ubicación en la que se extrae Redistributable Managed File Transfer Agent.

   Nota: Si el agente redistribuible se va a ejecutar como un servicio Windows, es necesario establecer la variable de entorno **BFG_DATA** en el entorno del sistema para que el servicio funcione.

   -n *nombre de instalación*
   Este parámetro se utiliza para especificar el nombre de una instalación de IBM MQ o un nombre exclusivo.

   Ejemplos de situaciones en las que posiblemente deseará utilizar este parámetro son:

   – Si desea probar rápidamente una nueva función o característica utilizando el paquete redistribuible con la configuración existente en la que los agentes se han configurado para que se conecten al gestor de colas solo en modalidad de clientes. (Tenga en cuenta que este

parámetro no se aplica a ningún agente que esté configurado para conectarse a un gestor de colas en modalidad de enlaces).

- Si va a realizar una migración desde una instalación de Managed File Transfer estándar a un paquete de Redistributable Managed File Transfer Agent y desea utilizar la misma configuración que la que ha creado la instalación estándar. Este es el caso en el que se ha instalado Managed File Transfer estándar, pero se está conectando a un gestor de colas de agente que se ejecuta en otra máquina.

La variable de nombre de instalación predeterminada es **BFG_INSTALLATION_NAME**.

Para obtener más información sobre el mandato **fteCreateEnvironment**, consulte fteCreateEnvironment (configuración de entorno para Redistributable Managed File Transfer Agent).

También puede establecer la variable de entorno *BFG_DATA* con la ubicación de la vía de acceso a datos:

```
BFG_DATA=Datapath location
```

Antes de crear, iniciar y parar un agente, o cualquier otro mandato, hay que asegurarse de que la variable *BFG_DATA* esté establecida a la ubicación correcta de la ruta de datos.

2. Configure la conectividad de IBM MQ.
   a) Configure el gestor de colas de coordinación con el mandato **fteSetupCoordination**.

   El mandato **fteSetupCoordination** crea la configuración necesaria para los gestores de colas de coordinación y los directorios necesarios para la configuración adicional. Redistributable Managed File Transfer Agent funciona en modo cliente, por lo que hay que proporcionar parámetros adicionales en este mandato para evitar un error, puesto que el modo de enlaces no está soportado.

   ```
   fteSetupCoordination -coordinationQMgr PRMFTDEMO2
                -coordinationQMgrHost 9.121.59.233 -coordinationQMgrPort 3002
                -coordinationQMgrChannel SYSTEM.DEF.SVRCONN
   ```

   Si desea más detalles y los pasos para utilizar el mandato **fteSetupCoordination**, consulte fteSetupCoordination. Para obtener información sobre cómo configurar el gestor de colas de coordinación, consulte “Configuración del gestor de colas de coordinación para MFT” en la página 801.

   b) Cree y configure el gestor de colas de mandatos:

   ```
   fteSetupCommands -p PRMFTDEMO2 -connectionQMgrHost 9.121.59.233
                -connectionQMgrPort 3002 -connectionQMgrChannel SYSTEM.DEF.SVRCONN
                -connectionQMgr PRMFTDEMO2 -f
   ```

   Para obtener más detalles y los pasos para utilizar el mandato **fteSetupCommands**, consulte fteSetupCommands: crear el archivo command.properties de MFT.

3. Cree una definición del agente de MFT para un punto final.

   ```
   fteCreateAgent -p PRMFTDEMO2 -agentQMgrHost 9.121.59.233
              -agentQMgrPort 3002 -agentQMgrChannel SYSTEM.DEF.SVRCONN
              -agentName AGENT.TRI.BANK -agentQMgr PRMFTDEMO2 -f
   ```

   Si desea más información sobre cómo utilizar el mandato **fteCreateAgent** para configurar un agente y el gestor de colas de agente, consulte fteCreateAgent.

   Nota: Debe utilizar los mandatos MQSC que se visualizan como parte de la salida del mandato para definir los objetos de agente en el gestor de colas del agente; de lo contrario, las instrucciones del paso “4” en la página 761 no funcionarán.

   En los pasos “2” en la página 761 y “3” en la página 761 de cada agente, se crean definiciones de cola y tema en el gestor de colas del agente.

4. Arranque el agente y estará listo para transferir archivos.

   ```
   fteStartAgent -p PRMFTDEMO2 AGENT.TRI.BANK
   ```

Puede verificar el estado del agente ejecutando el mandato siguiente:

```
fteListAgents
```

Si desea más detalles sobre cómo utilizar el mandato **fteListAgents**, consulte fteListAgents.

Qué hacer a continuación

Si desea configurar Redistributable Managed File Transfer Logger, complete los pasos de "Creación de la configuración inicial para Redistributable Managed File Transfer Logger" en la página 762.

Conceptos relacionados

"Configuración de Managed File Transfer" en la página 755
Puede configurar las características de Managed File Transfer después de la instalación.

"Opciones de configuración de MFT en Multiplatforms" en la página 755
Managed File Transfer proporciona un conjunto de archivos de propiedades que contienen información clave sobre la configuración y son necesarios para la operación. Estos archivos de propiedades están en el directorio de configuración que ha definido al instalar el producto.

Referencia relacionada

fteCreateTransfer: iniciar una nueva transferencia de archivos

Windows Linux Creación de la configuración inicial para Redistributable Managed File Transfer Logger

Puede configurar un Managed File Transfer Logger de tipo FILE para conectarse a un gestor de colas de coordinación en modalidad de cliente.

Antes de empezar

Asegúrese de haber descargado y extraído el contenido del paquete del Redistributable Managed File Transfer Agent. A partir de IBM MQ 9.3.0, este paquete incluye también Redistributable Managed File Transfer Logger. Para obtener más información, consulte "Descarga y configuración de Redistributable Managed File Transfer components" en la página 758.

Acerca de esta tarea

Redistributable Managed File Transfer Agent y Redistributable Managed File Transfer Logger comparten el mismo entorno. Una vez que se ha creado este entorno y se ha configurado la conectividad de IBM MQ, puede crear e iniciar el registrador.

Procedimiento

1. Asegúrese de que el entorno compartido para Redistributable Managed File Transfer Agent y Redistributable Managed File Transfer Logger se haya creado tal como se describe en el paso "1" en la página 760 y que se haya configurado la conectividad de IBM MQ tal como se describe en el paso "2" en la página 761 de "Creación de la configuración inicial para Redistributable Managed File Transfer Agent" en la página 760.

   .
2. Crear un registrador de archivos mediante el mandato **fteCreateLogger**.

   Por ejemplo:

   ```
   fteCreateLogger FILELOGGER -loggerType FILE -loggerQMgr PRMFTDEMO2
   -loggerQMgrHost 9.121.59.233 -loggerQMgrPort 3003 -loggerQMgrChannel SYSTEM.DEF.SVRCONN
   -fileSize 20MB -fileCount 10 -fileLoggerMode CIRCULAR
   ```

   Para obtener más información sobre cómo utilizar el mandato **fteCreateLogger**, consulte fteCreateLogger.
3. Inicie el registrador mediante el mandato **fteStartLogger**.

Para obtener más información sobre el mandato **fteStartLogger**, consulte fteStartLogger.

Conceptos relacionados

“Configuración de Managed File Transfer” en la página 755
Puede configurar las características de Managed File Transfer después de la instalación.

“Opciones de configuración de MFT en Multiplatforms” en la página 755
Managed File Transfer proporciona un conjunto de archivos de propiedades que contienen información clave sobre la configuración y son necesarios para la operación. Estos archivos de propiedades están en el directorio de configuración que ha definido al instalar el producto.

Windows Linux Actualización de Redistributable Managed File Transfer components

Puede actualizar el Redistributable Managed File Transfer components descargando un nuevo Redistributable Managed File Transfer package.

Antes de empezar

Para obtener información sobre los términos de licencia redistribuible para Redistributable Managed File Transfer Agent y Redistributable Managed File Transfer Logger , consulte Componentes redistribuibles deIBM MQ.

Nota: Advanced Message Security no está soportado con Redistributable Managed File Transfer package.

Acerca de esta tarea

Si ya ha instalado el Redistributable Managed File Transfer components, puede actualizarlos descargando un nuevo paquete redistribuible y extrayendo el contenido en la misma ubicación.

Procedimiento

1. Descargue el Paquete de agente IBM MQ redistribuible Managed File Transfer para el sistema operativo desde Fix Central.
2. Detenga todos los agentes de Managed File Transfer y el registrador espere a que se complete cualquier mandato Managed File Transfer en ejecución.
3. Actualice los archivos para la instalación existente de Redistributable Managed File Transfer components extrayendo el contenido del nuevo paquete redistribuible que ha descargado en el mismo directorio que el que ya tiene instalado Redistributable Managed File Transfer components .

z/OS Creating an MFT Agent or Logger command data set

You can create a PDSE data set of commands from the Managed File Transfer command template data set for a specific Managed File Transfer Agent or Managed File Transfer Logger for a specific coordination.

About this task

Complete the following steps:

Procedure

1. Make a copy of the MFT command template PDSE library data set SCSQFCMD.

   SCSQFCMD must be copied into a new library, for example *prefix.agent*.JCL. You can use an updated version of the SCSQFCMD(BFGCOPY) member with the following replacements:

   - Replace *++supplied-library++* with the fully qualified name of the SCSQFCMD PDSE.

- z/OS Replace *++service-library++* with the fully qualified name of the new MFT command PDSE library data set. The *++service-library++* is the output data set for the agent or logger service that is created.

2. For the new MFT command PDSE library data set, edit the member BFGCUSTM, which is a JCL script to customize the commands for the agent or logger. Each variable is specified in the format: *++variable name++*, which you must replace with its required value. For a description of the various JCL variables, see "z/OS JCL variables" on page 776. The BFGSTDIN DD statement defines variables in three categories: Variables, Properties, and Environment. The statement has the following format:

```
[Variables]
variable1=value1
variable2=value2
....
variableN=valueN
[Properties]
property1=property value1
property2=property value2
...
propertyN=property valueN
[Environment]
custom_variable1=value1
custom_variable2=value2
....
custom_variableN=valueN
```

   Variables define the set of setup and environment variables that are required for each command.

   Properties define overrides for the MFT configuration properties. You can add agent and logger properties as required to customize the agent or logger for your environment. For a list of all properties, see "Configuration properties files" on page 786. This facility is provided to save having to access the MFT configuration properties files, which are maintained as z/OS UNIX System Services files.

   Environment defines any additionally required custom environment variables.

3. Submit job BFGCUSTM for the new MFT command PDSE library data set. This job generates the set of JCL commands, as new members of the PDSE, appropriate for the agent or logger. For a full list of the commands, see "z/OS agent and logger command JCL scripts" on page 779.

   Job BFGCUSTM updates the library containing the JCL which includes a DD statement with DISP=OLD. You must exit the editor after submission to allow the job to execute.

   Examine the output job log to check that the JCL script ran successfully. If there are any failures, correct them and submit the BFGCUSTM job again.

   The BFGCUSTM JCL script also updates the z/OS UNIX System Services MFT configuration properties files as necessary to keep the files in step. If the configuration defined by the CoordinationQMgr property does not exist, warning messages are output and you must run the generated BFGCFCR and BFGCMCR jobs to create the configuration properties files. You must run BFGAGCR for an agent, and BFGLGCRS for a logger edit. If the specified configuration already exists, the configuration is updated with any properties as defined in the BFTCUSTM JCL script.

Related concepts

"MFT configuration options on z/OS" on page 757
The Managed File Transfer configuration options on z/OS are the same as the options for distributed platforms.

Related tasks

"Updating an existing MFT Agent or Logger command data set on z/OS" on page 775
You can update an Managed File Transfer command PDSE library data set that is created from the Managed File Transfer command template data set.

z/OS Configuring Managed File Transfer for z/OS

Managed File Transfer for z/OS requires customization to enable the component to operate correctly.

About this task

You need to:

1. Edit a PDSE member to specify configuration data
2. Define the coordination queue manager.
3. Define the command queue manager
4. Configure one or more agents
5. Optionally: configure a logger task to store data in Db2

The sequence of tasks you need to perform is detailed in the following topics.

Related concepts

"Reviewing the MFT configuration" on page 765
You need to review the configuration of your system before you begin.

Related tasks

Installing IBM MQ Advanced for z/OS

z/OS Reviewing the MFT configuration

You need to review the configuration of your system before you begin.

Managed File Transfer (MFT) requires one or more queue managers to act in the following roles for each defined MFT configuration:

- A coordination queue manager, which maintains information on the status of each agent in the configuration published to a topic on the coordinator.
- One or more command or connection queue managers that act as the entry point to the IBM MQ network for MFT commands.
- One or more agent queue managers that provide the communication between an MFT agent and the IBM MQ network.

Each of the above roles can be performed by a separate queue manager, or you can combine the roles, so that, in the simplest configuration, all roles are performed by a single queue manager.

If you are adding a z/OS queue manager to an existing MFT environment you need to define connectivity between the z/OS queue manager and the other queue managers in the configuration. You can achieve this with manually defined transmission queues, or by the use of clustering.

Each MFT agent communicates with a single queue manager. If multiple agents communicate with the same queue manager, then the agent queue manager will have multiple queues defined for each agent:

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

Note that you can define generic security profiles, where you use a profile such as `SYSTEM.FTE.COMMAND.*`, or you can define specific profiles for each agent.

Related concepts

“Antes de empezar a configurar MFT para z/OS” on page 766
La configuración de Managed File Transfer (MFT) utiliza archivos en conjuntos de datos z/OS UNIX System Services (z/OS UNIX) y PDSE.

Related reference

MFT system queues and the system topic

z/OS Antes de empezar a configurar MFT para z/OS

La configuración de Managed File Transfer (MFT) utiliza archivos en conjuntos de datos z/OS UNIX System Services (z/OS UNIX) y PDSE.

La mayor parte de la configuración y de las operaciones se realizan utilizando JCL desde un PDSE y debe estar familiarizado con el trabajo en un entorno de (z/OS UNIX).

Puede acceder a OMVS desde ISPF o puede utilizar una sesión de tipo Telnet utilizando mandatos en su estación de trabajo, por ejemplo, Telnet Putty o SSH.

Si utiliza OMVS desde ISPF, puede utilizar los mandatos ISPF estándar para editar y examinar, **oedit** y **obrowse**.

Debe estar familiarizado con los siguientes mandatos de (z/OS UNIX):

Tabla 41. Mandatos comunes de z/OS UNIX

Mandato	Función
chmod xxx víaacceso	Cambiar permisos de acceso a archivos.
df -k víaacceso	Indica la cantidad de espacio libre que queda en el sistema de archivos. -k indica el espacio libre en KB.
du -kt víaacceso	Indica los tamaños de los directorios de la vía de acceso. Tamaño notificado en KB.
find víaacceso -name xxx	Busca el archivo con el nombre xxxx en el directorio de la vía de acceso. xxx distingue entre mayúsculas y minúsculas y puede ser similar a *zzz.
ls -ltrd directorio	Lista información acerca del directorio especificado, en lugar de los archivos del directorio.
ls -ltr path	Lista información sobre los archivos en la vía de acceso.
obrowse nombrearchivo	Examine el nombre de archivo.
oedit nombrearchivo	Edite un archivo en OMVS.

Revise los elementos de la tabla siguiente y complete la tabla con las entradas adecuadas para su empresa. Necesitará estos valores cuando edite el miembro BFGCUSTM.

Tabla 42. Parámetros necesarios para el miembro BFGCUSTM

Nombre	Datos de ejemplo	Comentarios
ADMIN_JOB1		Tarjeta de trabajo. Todos los trabajos se generan con la misma tarjeta JCL.

Tabla 42. Parámetros necesarios para el miembro BFGCUSTM (continuación)		
Nombre	**Datos de ejemplo**	**Comentarios**
armELEMENT	Si se está utilizando ARM, utilice el valor ARM ELEMENT especificado en la política de ARM para este agente o registrador. Si no se está utilizando ARM, establezca este parámetro en blanco; por ejemplo, armELEMENT=	
armELEMTYPE	Si se está utilizando ARM, utilice el valor ARM ELEMTYPE especificado en la política de ARM. Por ejemplo, armELEMTYPE=SYSBFGAG para un agente o armELEMTYPE=SYSBFGLG para un registrador. Si no se está utilizando ARM, establezca este parámetro en blanco; por ejemplo, armELEMTYPE=	
BFG_DATA		Complete según sea necesario
BFG_GROUP_NAME	MQM	
BFG_JAVA_HOME	`/java/java71_bit64_GA/J7.1_64/`	
BFG_JVM_PROPERTIES		Complete según sea necesario
BFG_PROD	`/mqm/V9R2M0/mqft`	La vía de acceso completa al directorio `mqft` bajo el directorio IBM MQ for z/OS UNIX System Services Components .
BFG_WTO	SÍ	Para obtener un mensaje MFT en el syslog.
CLEAN_AGENT_PROPS	-trs	Este parámetro especifica las opciones que se utilizarán para borrar un agente cuando se ejecuta el miembro BFGAGCL. Para obtener más información sobre los valores válidos para este parámetro, consulte fteCleanAgent: limpiar un agente de MFT.
coordinationQMgr	MQPV	Configuración obligatoria
CREDENTIAL_PATH		Se utiliza en la migración
_HLQ de Db2	SYS2.Db2.V10	
DB_PROPS_PATH		Se utiliza en la migración
FTE_CONFIG		Se utiliza en la migración
JOBCARD1		Esta es la tarjeta de trabajo para las tareas de larga ejecución, agentes y registradores.

Tabla 42. Parámetros necesarios para el miembro BFGCUSTM (continuación)		
Nombre	**Datos de ejemplo**	**Comentarios**
LIBRARY	SCEN.FTE.JCL	Nombre de MFT de PDSE. Necesita una copia de cada tarea de registro o agente.
MQ_HLQ	El calificador de alto nivel para los conjuntos de datos de IBM MQ. Por ejemplo, `MQM.V920`	
MQ_LANG	E	
MQ_PATH	`/mqm/V9R2M0`	La vía de acceso completa del directorio a la instalación de IBM MQ for z/OS UNIX System Services Components.
NOMBRE	AGENT1	
OUTPUT_CLASS	*	
PATH	`bin:/usr/bin:/usr/sbin`	
productId	ADVANCEDVUE	Este parámetro se utiliza para establecer el tipo de producto para el cual el uso de Managed File Transfer se va a registrar. Para obtener más información sobre los valores válidos para este parámetro, consulte fteSetProductId: establecer ID de producto de registro de SCRT de z/OS.
QMGR	MQPV	
SERVICE_TYPE	AGENT o LOGGER	
TMPDIR	`/tmp`	Lee y graba en la vía de acceso de z/OS UNIX accesible los archivos temporales.

Además, deberá revisar las variables siguientes y suministrar los valores donde sean necesarios:

- coordinationQMgrHost=
- coordinationQMgrPort=
- coordinationQMgrChannel=
- connectionQMgr=
- connectionQMgrHost=
- connectionQMgrPort=
- connectionQMgrChannel=

Estas propiedades son comunes para AGENT o LOGGER.

Nota: El host, el puerto y el canal son necesarios para la conexión de cliente, pero deben permanecer en blanco para una conexión de enlaces en la máquina local.

Conceptos relacionados

"Items to check" en la página 769
Ensure that you have enough disk space, a directory for storing data, and that the requisite files exist.

“Editing member BFGCUSTM” en la página 771
You must edit member BFGCUSTM, and enter the values for the parameters that your enterprise uses, before you run the job.

z/OS Items to check

Ensure that you have enough disk space, a directory for storing data, and that the requisite files exist.

Check you have enough disk space

Check that you have enough disk space available on the file system where you are going to store the configuration specific files.

If an agent trace is enabled then by default it can use 100 MB of disk space.

The configuration files themselves are small, only a few KB in size.

If you are planing on using two agents and a logger then you need at least 300 MB. You can use the command **df -k** `path`, where `path` is the location of the installation specific files. This gives the available and total space in KB.

300 MB is 307,200 KB so you should allow for at least 310,000 KB

Create and check the directory for storing Managed File Transfer data

You need a directory for storing the Managed File Transfer (MFT) data.

Check you have enough space in the file system `df -k /var`. This file system should have at least 310,000 KB available.

If you have not created this file system, use the **mkdir** command; for example **mkdir /var/mft**.

Display what permissions users have on this directory, using the command **ls -ltrd /var/mft**.

If the owner or group is not correct, use the command **chown owner:group /var/mft**.

If permissions for the group are not correct, use the following command to give the owner and the group read, write, and execute permissions. Note that the following command also gives all users read and execute permissions **chmod 775 /var/mft**.

Check the files exist and you have access to them

Use the **ls -ltr** command for the files you will be using during customization. For example:

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

gives

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

where the `drwxr-xr-x` means

d
: This is a directory.

rwx
: The owner *SYSTASK* has read, write and execute access to the directory.

r-x
: People in the group *TSOUSER* can read and execute files in the directory.

r-x
: Universal access, that is, anyone can read or execute files in the directory.

Check the files specified in:

Table 43. Access required by users to specific files	
Path	**Access required by users doing the configuration**
BFG_JAVA_HOME	Read and execute
/tmp	Read and write
BFG_PROD	Read
BFG_DATA	Write
MQ_PATH	Read

Related concepts

"Antes de empezar a configurar MFT para z/OS" on page 766
La configuración de Managed File Transfer (MFT) utiliza archivos en conjuntos de datos z/OS UNIX System Services (z/OS UNIX) y PDSE.

"Common MFT for z/OS configurations" on page 770
An overview of the different Managed File Transfer configurations

z/OS Common MFT for z/OS configurations

An overview of the different Managed File Transfer configurations

Managed File Transfer uses agents attached to a queue manager for transferring data.

MFT puede utilizar varios gestores de colas:

- Uno o más gestores de colas para transferir los datos.
- Un gestor de colas de mandatos que emite solicitudes. Por ejemplo, se envía una solicitud para iniciar una transferencia a este gestor de colas y los mandatos asociados se direccionan a los agentes MFT.
- Un gestor de colas de coordinación que gestiona el trabajo.

Hay tres configuraciones comunes de Managed File Transfer (MFT):

1. Un único gestor de colas con uno o más agentes utilizando conexiones locales. Éste se puede utilizar para poner el contenido de un conjunto de datos en colas de IBM MQ.
2. Un único gestor de colas con un cliente MFT en una máquina distribuida utilizando enlaces de cliente.
3. Dos gestores de colas conectados por canales y uno o más agentes en cada máquina. Estos agentes pueden ser enlaces de cliente o locales.

Tenga en cuenta las siguientes cuestiones:

1. MFT se escribe en Java, con algunos scripts de shell y JCL para configurar y operar MFT.
2. El estado y la actividad de Db2 pueden registrarse y almacenarse en tablas de Db2.
3. La persona que configura MFT debe estar familiarizado con z/OS UNIX System Services (z/OS UNIX). Por ejemplo:
   - La estructura de directorios con archivos con nombres como `/u/userID/myfile.txt2`
   - comandos de z/OS UNIX como, por ejemplo:

     cd (cambiar directorio)
     ls (Lista)
     chmod (cambiar permisos de archivos)
     chown (cambiar la propiedad del archivo o los grupos que pueden acceder al archivo o directorio)
4. Los siguientes productos son necesarios en z/OS UNIX para poder configurar y ejecutar MFT:
   - Java; por ejemplo, `/java/java71_bit64_GA/J7.1_64/`
   - IBM MQ V920, por ejemplo `/mqm/V9R2M0`.

- Bibliotecas JDBC de Db2, si desea utilizar Db2 para el estado y el historial; por ejemplo, `/db2/db2v12/jdbc/lib`

You need a coordination queue manager. However, you can use the same queue manager to run agents, to process commands, and for coordination. If you are using multiple queue managers, you must pick one to act as the coordinator.

Check your IBM MQ connectivity

If you have an existing MFT coordinator queue manager, you need connectivity between the queue manager where you are doing the configuration, and the coordinating and command queue managers.

z/OS Copy SCSQFCMD to create a JCL library

You need to create a JCL library for each agent and logger. The JCL contains the configuration and jobs used to create and run the agent or logger.

For each agent and logger create a copy of the IBM supplied SCSQFCMD library by editing and running the BFGCOPY member.

This library is used to define the configuration for the agent or logger and, after customization, contains jobs that can be used to create the required Managed File Transfer configuration and agent or logger.

You create member BFGCUSTM as part of this process.

Note: If you are familiar with z/OS UNIX commands, you can configure z/OS with the same commands that you use on other platforms.

Related concepts

“Common MFT for z/OS configurations” on page 770
An overview of the different Managed File Transfer configurations

“Editing member BFGCUSTM” on page 771
You must edit member BFGCUSTM, and enter the values for the parameters that your enterprise uses, before you run the job.

z/OS Editing member BFGCUSTM

You must edit member BFGCUSTM, and enter the values for the parameters that your enterprise uses, before you run the job.

See Parameters needed for member BFGCUSTM, for a list of the parameters requiring specific values.

Además, deberá revisar las variables siguientes y suministrar los valores donde sean necesarios:

- coordinationQMgrHost=
- coordinationQMgrPort=
- coordinationQMgrChannel=
- connectionQMgr=
- connectionQMgrHost=
- connectionQMgrPort=
- connectionQMgrChannel=

Estas propiedades son comunes para AGENT o LOGGER.

Note: El host, el puerto y el canal son necesarios para la conexión de cliente, pero deben permanecer en blanco para una conexión de enlaces en la máquina local.

If this is the first queue manager in your Managed File Transfer environment, and you want to use the same queue manager for coordination, commands, and running agents, set the values to the local queue manager name.

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

where MQPV is your local queue manager name.

Submit the job, which updates the PDSE, and creates a directory structure under the specified path.

Note that this job requires exclusive use, so you need to stop using the PSDE while the job runs.

Tip: Whenever you submit job BFGCUSTM, the job replaces all the JCL files. You should rename each member you change.

Related concepts

"Antes de empezar a configurar MFT para z/OS" on page 766
La configuración de Managed File Transfer (MFT) utiliza archivos en conjuntos de datos z/OS UNIX System Services (z/OS UNIX) y PDSE.

"Creating an agent" on page 774
You need to copy the PDSE to make the agent-specific PDSE, for example *user.MFT.AGENT1*. Copy the PDSE from a previous agent or logger configuration, if they exist. If this is your first configuration, copy the PDSE supplied with MFT.

z/OS Defining the coordination queue manager

Managed File Transfer requires a queue manager to be created that acts as the coordination queue manager.

Depending on the configuration that you have chosen, this queue manager is on the local MVS system, or on another machine. In the former case, the connections to it are bindings connections and in the latter case, they are client connections.

After you have run the configuration step successfully there are configured members in the PDSE.

Member BFGCFCR defines the coordination queue manager, and this job:

1. Creates a directory structure in the Managed File Transfer (MFT) directory, and creates configuration files.
2. Runs CSQUTIL to define IBM MQ resources.

If the coordination queue manager is on a remote machine then this job step fails.

Member BCFCFCR creates files in z/OS UNIX System Services and creates MQ definitions. This job:

1. Creates an MFT topic,
2. Creates an MFT queue
3. Alters *NAMELIST(SYSTEM.QPUBSUB.QUEUE.NAMELIST)* to be *NAMES(SYSTEM.BROKER.DEFAULT.STREAM, SYSTEM.BROKER.ADMIN.STREAM, SYSTEM.FTE)*
4. Performs *ALTER QMGR PSMODE(ENABLED)*

   A *DISPLAY NAMELIST(SYSTEM.QPUBSUB.QUEUE.NAMELIST)* command is issued before doing the alter. If your NAMLIST is not the default, you should alter your name list to add SYSTEM.FTE to your namelist

Rename member BCFCFCR with your own prefix, for example, CCPCFCR, because re customizing this file replaces it.

Edit this renamed member by inserting the name of your credentials file. For example:

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

Save and submit the job. Note that if you need to resubmit the job, you need to add the *-f* option.

When this job runs it lists the IBM MQ resources it creates. You need to protect these resources.

```
DEFINE TOPIC('SYSTEM.FTE') TOPICSTR('SYSTEM.FTE') REPLACE
ALTER TOPIC('SYSTEM.FTE') NPMSGDLV(ALLAVAIL) PMSGDLV(ALLAVAIL)
DEFINE QLOCAL(SYSTEM.FTE) LIKE(SYSTEM.BROKER.DEFAULT.STREAM) REPLACE
ALTER QLOCAL(SYSTEM.FTE) DESCR('Stream for MFT Pub/Sub interface')
* Altering namelist: SYSTEM.QPUBSUB.QUEUE.NAMELIST
* Value prior to alteration:
DISPLAY NAMELIST(SYSTEM.QPUBSUB.QUEUE.NAMELIST)
ALTER NAMELIST(SYSTEM.QPUBSUB.QUEUE.NAMELIST) +
NAMES(SYSTEM.BROKER.DEFAULT.STREAM+
,SYSTEM.BROKER.ADMIN.STREAM,SYSTEM.FTE)
* Altering PSMODE. Value prior to alteration:
DISPLAY QMGR PSMODE
ALTER QMGR PSMODE(ENABLED)
```

Related tasks

“Defining the command queue manager” on page 773
You can either use the same queue manager as the coordination and command queue managers, or create a new command queue manager.

z/OS Defining the command queue manager

You can either use the same queue manager as the coordination and command queue managers, or create a new command queue manager.

About this task

You must have a command queue manager, however, you can use the same queue manager for the coordination and command queue managers. Otherwise, you need to create a new command queue manager. This can be on the same machine as the coordination queue manager, but does not have to be.

Procedure

1. Rename member BFGCMCR with your own prefix, for example, CCPCMCR.

   You must rename BFGCMCR because re-customizing this file replaces it.
2. Edit the renamed member by inserting the name of your credentials file.

   For example:

   ```
   %BFGCMD CMD=fteSetupCommands +
   -credentialsFile //'<MFTCredentialsDataSet(MemberName)>' +
   ```
3. Save and submit the job.

   Note that if you need to resubmit the job, you need to add the *-f* option.

   This queue manager is used for commands such as **ftePingAgent**.
4. Review this member, submit it, and review the output.

What to do next

See “Creating an agent” on page 774 for information on how you create an agent.

Related concepts

“Defining the coordination queue manager” on page 772
Managed File Transfer requires a queue manager to be created that acts as the coordination queue manager.

Related tasks

Configuring MQMFTCredentials.xml

Related reference

MFT credentials file format

z/OS Creating an agent

You need to copy the PDSE to make the agent-specific PDSE, for example *user.MFT.AGENT1*. Copy the PDSE from a previous agent or logger configuration, if they exist. If this is your first configuration, copy the PDSE supplied with MFT.

Review member BFGCUSTM and if you need to use a different credentials file, create one.

Much of the content remains the same from the customization detailed in "Editing member BFGCUSTM" on page 771.

You need to change:

- //SYSEXEC DD DSN=SCEN.FTE.JCL.AGENT1
- LIBRARY to match the agent PDSE
- SERVICE_TYPE=AGENT
- NAME to be the name of the agent (matching the PDSE) JOBCARD
- Change BFG_JVM_PROPERTIES="-Xmx1024M"

Submit this job, remembering that the job requires exclusive access to the data set.

The jobs for the agent all have names of the form *BFGAG**

Rename member *BFGAGCR*. This job updates files in the Managed File Transfer directory and uses CSQUTIL to create agent specific queues in the local queue manager. Specify the name of your credentials file, for example, `-credentialsFile //'SCEN.FTE.JCL.VB(CREDOLD)`. If you do not specify the name, the job to start the agent does not use a credentials file.

Check the output to ensure that the process has run successfully.

Tip: Copy the path name of the *agent.properties* file from the output of the job to a member in the PDSE for the agent.

For example, copy `/u/userid/fte/wmqmft/mqft/config/MQPA/agents/AGENT1/agent.properties` into member AGENT.

This is useful if you need to display the properties file, and add the line `/u/userid/fte/wmqmft/mqft/logs/MQPA/agents/AGENT1/logs`.

This is where trace files are stored.

Related concepts

"Defining the coordination queue manager" on page 772
Managed File Transfer requires a queue manager to be created that acts as the coordination queue manager.

"Using the agent" on page 774
How you use various commands to ensure that the agent is working correctly.

Related tasks

"Defining the command queue manager" on page 773
You can either use the same queue manager as the coordination and command queue managers, or create a new command queue manager.

z/OS Using the agent

How you use various commands to ensure that the agent is working correctly.

Start the agent

Rename member BFGAGST, review the member, and submit the job.

If this works you receive message BFGAG0059I: The agent has been successfully started.

Display the active agent(s)

Rename member BFGAGLI, review the member and submit the job which uses the coordinating queue manager.

You must resolve any connectivity problems

Ping the agent to check it is working

Rename member BFGAGPI, review the member and submit the job which uses the command queue manager.

You must resolve any connectivity problems

Carry out a test transfer

See "Performing a verification transfer" on page 781 for further information.

Stop the agent

Rename member BFGAGSP, review the member and submit the job.

Restart the agent using the member BFGAGST.

Related concepts

"Creating an agent" on page 774
You need to copy the PDSE to make the agent-specific PDSE, for example *user.MFT.AGENT1*. Copy the PDSE from a previous agent or logger configuration, if they exist. If this is your first configuration, copy the PDSE supplied with MFT.

z/OS Updating an existing MFT Agent or Logger command data set on z/OS

You can update an Managed File Transfer command PDSE library data set that is created from the Managed File Transfer command template data set.

Procedure

1. Edit the BFGCUSTM JCL script member and update variables and properties in the BFGSTDIN DD statement.

   If you want to remove a property that was previously defined, set its value to blank, instead of removing the entry. When the BFGCUSTM JCL script is run, the specified properties are applied as an update to the actual agent and logger z/OS UNIX System Services properties files; setting a property to a blank value indicates that the property is to be removed
2. Submit job BFGCUSTM. This job generates the set of JCL commands again, appropriate for the agent or logger. For a full list of the commands, see "z/OS agent and logger command JCL scripts" on page 779. Examine the output job log to check that the JCL script ran successfully. If there are any failures, correct them and submit the BFGCUSTM job again.

Results

You can modify the generated JCL scripts and add your own logic. However, be careful when you run BFGCUSTM again because you might overwrite the custom logic.

Related concepts

"MFT configuration options on z/OS" on page 757
The Managed File Transfer configuration options on z/OS are the same as the options for distributed platforms.

Related tasks

"Creating an MFT Agent or Logger command data set" on page 763
You can create a PDSE data set of commands from the Managed File Transfer command template data set for a specific Managed File Transfer Agent or Managed File Transfer Logger for a specific coordination.

z/OS *z/OS JCL variables*

You can use substitution values, JCL variables, and configuration properties in the BFGCUSTM script.

The following table lists the substitution values for the BFGCUSTM JCL script in an MFT command PDSE library data set. You must replace these substitution values with suitable values before you submit the BFGCUSTM job.

Table 44. Substitution values for the BFGCUSTM JCL script

Substitution variable	**Value**
++library++	The data set name of the containing MFT command PDSE library.
++bfg_java_home++	The location of your Java installation.
++mq_path++	The path to the IBM MQ for z/OS UNIX System Services Components directory. For example, `/mqm/V9R2M0`. This is used to give the full path to the MFT installation, for example, `/mqm/V9R2M0/mqft`.

The following table describes the environment variables for the BFGSTDIN DD statement for the BFGCUSTM JCL script, in an MFT command PDSE library data set (in the [Variables] section). You must replace all variables that are specified with substitution values (that is, values enclosed in two plus signs, ++) with suitable values before you submit the BFGCUSTM job.

Table 45. Environment variables

Environment variable	**Value**
LIBRARY	The data set name of the containing MFT command PDSE library.
TMPDIR	z/OS UNIX System Services directory for temporary files.
BFG_PROD	The full path to the `mqft` directory under the IBM MQ for z/OS UNIX System Services Components directory; For example: `/mqm/V9R2M0/mqft`.
BFG_DATA	The location of the data directory for Managed File Transfer for z/OS, which is the path to *DATA_DIR*.
BFG_JAVA_HOME	The location of your Java installation.
BFG_JVM_PROPERTIES	Optional. Sets a value for the BFG_JVM_PROPERTIES environment variable. These properties are passed to the Java virtual machine.

Table 45. Environment variables (continued)

Environment variable	Value
BFG_GROUP_NAME	The mqm file group is typically associated with MFT configuration data files and commands. Consequently, all users who are members in the mqm group can access and make changes to the MFT configuration. For more information, see File system permissions for MFT in IBM MQ. For a z/OS system, a file group is a z/OS UNIX System Services (z/OS UNIX) filesystem entity, and the mqm file group is not necessarily defined. You can associate a z/OS UNIX filesystem group for MFT configuration data files by using the BFG_GROUP_NAME environment variable. For example, at the z/OS UNIX shell prompt use: `export BFG_GROUP_NAME=FTEGB` which defines a group *FTEGB* to be associated with any subsequently created configuration files for the current z/OS UNIX session. You can set BFG_GROUP_NAME to a blank value, or remove it. **Note:** When running BFGCUSTM for the first time, if the MFT configuration is to be used by multiple user IDs, it is important that BFG_GROUP_NAME is set to a group accessible to all required user ID's. If BFGCUSTM is run again, then BFG_GROUP_NAME must not be changed (otherwise, the z/OS UNIX group file permissions for all files and directories in the directory referenced by BFG_DATA must also be changed to reflect the new BFG_GROUP_NAME setting).
BFG_WTO	z/OS logging is enabled when BFG_WTO is set to YES, ON, or TRUE. This controls whether messages that are written to the agent event log are also written to the z/OS operator log facility, which allows easier access for automation products when you run an agent from JCL. The routing code is Programmer Information (11) and the descriptor code is Informational (12).
SERVICE_TYPE	Specifies whether the MFT command library is for an agent or logger. The valid values are AGENT or LOGGER.
NAME	The name of the agent or logger for the SERVICE_TYPE value.
QMGR	The name of the local queue manager that is associated with the agent or logger for the SERVICE_TYPE value.

Table 45. Environment variables (continued)	
Environment variable	**Value**
OUTPUT_CLASS	The ouput class for SYSOUT data sets. Defaults to * which requests the same output class as the MSGCLASS parameter from the job statement.
MQ_PATH	The path to the IBM MQ for z/OS UNIX Components directory.
MQ_HLQ	The high-level qualifier for IBM MQ data sets.
MQ_LANG	The language that is required.
DB2_HLQ	Optional. High-level qualifier for Db2 data sets.
JOBCARD1	Header line 1 for a JCL command job.
JOBCARD2	Header line 2 for a JCL command job.
JOBCARD3	Header line 3 for a JCL command job.
ADMIN_JOB1	Header line 1 for an admin job.
ADMIN_JOB2	Header line 2 for an admin job.
ADMIN_JOB3	Header line 3 for an admin job.
FTE_CONFIG	Existing MFT configuration for migration. Set to a blank value if migration is not required.
CREDENTIAL_PATH	Path to credentials file for migration, for example `/u/user1/agent3`. Required for migration commands **BFGAGMG** and **BFGLGMG** JCL scripts only. Set to a blank value if migration is not required. Note also that
DB_PROPS_PATH	Specifies the database logger properties file for migration. This option is required only if the properties file does not use the following default name and path: `config_directory/coordination_qmgr/databaselogger.properties`. Set to a blank value if migration is not required.

The following table describes the mandatory MFT configuration properties for the BFGSTDIN DD statement for the BFGCUSTM JCL script in an MFT command PDSE library data set. You must replace properties specified with substitution values (that is, values enclosed in two plus signs, ++) with a suitable non-blank value before you submit the BFGCUSTM job. These properties define overrides for the MFT configuration properties. You can add agent and logger properties to customize agents or loggers for your environment. For a list of all properties, see “Configuration properties files” on page 786.

Table 46. Mandatory configuration properties for the BFGSTDIN DD statement	
Property	**Value**
coordinationQMgr	The name of the coordination queue manager for the configuration that the agent or logger is associated with.
coordinationQMgrHost	Optional. Host name of the system that the coordination queue manager is running on. If you leave the value for this property blank, a bindings mode connection is assumed.

Table 46. Mandatory configuration properties for the BFGSTDIN DD statement (continued)	
Property	**Value**
coordinationQMgrPort	Optional. Port number that the coordination queue manager is listening on. This parameter is used only if you also specify a non-blank value for the coordinationQMgrHost property.
coordinationQMgrChannel	Optional. Channel to use to connect to the coordination queue manager. This parameter is used only if you also specify a non-blank value for the coordinationQMgrHost property.
connectionQMgr	The name of the command queue manager for the configuration that the agent or logger is associated with.
connectionQMgrHost	Optional. Host name of the system that the command queue manager is running on. If you leave the value for this property blank, a bindings mode connection is assumed.
connectionQMgrPort	Optional. Port number that the command queue manager is listening on. This parameter is used only if you also specify a non-blank value for the connectionQMgrHost property.
connectionQMgrChannel	Optional. Channel to use to connect to the command queue manager. This parameter is used only if you also specify a non-blank value for the connectionQMgrHost property.

z/OS *z/OS agent and logger command JCL scripts*

The set of JCL commands available in an MFT command PDSE library data set.

Table 47. JCL commands available in an MFT command PDSE library data set	
Member	**Description or fte command line command**
BFGCOPY	Job to create a copy of this library
BFGCUSTM	Job to customize this library for agent or logger
BFGZCFCR	fteSetupCoordination
BFGZCMCR	fteSetupCommands: create the MFT command.properties file
BFGZAGCR	fteCreateAgent. Created only when you set the SERVICE_TYPE variable to AGENT.
BFGLGCRS	fteCreateLogger. Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGZAGST	fteStartAgent. Created only when you set the SERVICE_TYPE variable to AGENT.
BFGAGSTP	**fteStartAgent** procedure. Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZAGPI	ftePingAgent. Created only when you set the SERVICE_TYPE variable to AGENT.

Table 47. JCL commands available in an MFT command PDSE library data set (continued)	
Member	**Description or fte command line command**
BFGZAGSP	fteStopAgent. Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZLGST	fteStartLogger. Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGLGSTP	**fteStartLogger** procedure. Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGZLGSP	fteStopLogger. Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGZAGSH	fteShowAgentDetails. Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZLGSH	fteShowLoggerDetails. Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGZCFDF	fteChangeDefaultConfigurationOptions
BFGZAGCL	fteCleanAgent. Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZAGDE	fteDeleteAgent. Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZLGDE	fteDeleteLogger. Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGZPRSH	fteDisplayVersion
BFGZAGLI	fteListAgents. Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZMNLI	fteListMonitors
BFGZSTLI	fteListScheduledTransfers
BFGZTMLI	fteListTemplates
BFGXCROB	**fteObfuscate** sample
BFGZRAS	**fteRAS**
BFGZAGTC	fteSetAgentTraceLevel. Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZLGTC	fteSetLoggerTraceLevel. Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGXPRAN	**fteAnt** sample
BFGXTRCA	**fteCancelTransfer** sample
BFGXMNCR	**fteCreateMonitor** sample
BFGXTMCR	**fteCreateTemplate** sample
BFGXTRCR	**fteCreateTransfer** sample
BFGXMNDE	**fteDeleteMonitor** sample
BFGXSTDE	**fteDeleteScheduledTransfer** sample
BFGXTMDE	**fteDeleteTemplate** sample

Notes:

- The JCL, for commands that create MQSC or reference delete scripts, asks you to run a script, but the script has already been run by the job.
- BFGZRAS creates the BFGRAS member when the BGCUSTM job is run.

z/OS Performing a verification transfer

How you carry out a transfer to check that the product is working correctly.

Rename and edit member BFGTRCRS.

1. Add a `/*` before the `%BFGCMD CMD=fteCreateTransfer -h`
2. Remove the other comments in the member.
3. Specify the current agent name for `-sa` and `-da`
4. Save the JCL
5. Submit the JCL

This JCL connects to the command queue manager.

z/OS Configuración de una tarea de registro

La tarea de registro se ha de ejecutar en la misma imagen que el gestor de colas de coordinación. Puede registrar en Db2.

Creación de una tarea de registro

Copie el PDSE para que el registrador sea específico de PDSE. Por ejemplo, `user.MFT.LOGGER`.

Si tiene que utilizar un archivo de credenciales diferente, cree uno. Consulte Configuración de MQMFTCredentials.xml en z/OS.

Revise el miembro BFGCUSTM. Tenga en cuenta que gran parte del contenido continúa siendo el mismo de la personalización anterior.

Sin embargo, debe:

- Cambiar //SYSEXEC DD DSN=SCEN.FTE.JCL....
- Cambiar LIBRARY para que coincida con el PDSE del agente
- Cambiar QMGR al nombre del gestor de colas de coordinación
- Establecer SERVICE_TYPE=LOGGER
- Cambiar NAME para que sea el nombre del registrador (coincidente con PDSE)
- Revisar JOBCARD y cambiar el nombre de trabajo, de modo que el nombre sea diferente de los nombres de trabajo de los agentes.
- Revisar BFG_JVM_PROPERTIES="-Xmx1024M"

Si está utilizando el registrador de Db2, es conveniente crear un archivo, para que pueda capturar rastreos de Db2 que le ayuden a identificar problemas de Db2.

El nombre del archivo se especifica en las propiedades de la JVM, donde el contenido del archivo de rastreo de JDBC es similar al siguiente

```
db2.jcc.traceDirectory=/u/johndoe/fte
db2.jcc.traceFile=jccTrace1
db2.jcc.traceFileAppend=false
# turn on all traces
# db2.jcc.traceLevel=-1
# turn off all traces
db2.jcc.traceLevel=0
```

Establezca dos propiedades de JVM

```
BFG_JVM_PROPERTIES=-Ddb2.jcc.propertiesFile=/u/.../sql.properties
-Ddb2.jcc.ssid=DBCA
```

Donde `/u/.../sql.properties` es el nombre de su archivo de propiedades de rastreo de Db2 y *DBCA* es el nombre de su subsistema Db2.

Envíe el trabajo, teniendo en cuenta que el trabajo requiere acceso exclusivo al conjunto de datos. Los trabajos para el agente tienen todos nombres como *BFGLG**.

Registro en archivos

Para obtener más información sobre el registro en Db2, consulte “Creación de una tarea de registro al registrar en Db2” en la página 783

Cambie el nombre del miembro BFGLGCRS. Este trabajo actualiza los archivos en el directorio Managed File Transfer (MFT) y utiliza CSQUTIL para crear colas específicas del agente en el gestor de colas local.

El archivo original tiene el mandato `%BFGCMD CMD=fteCreateLogger -h`, el cual lista la sintaxis del mandato.

Para crear la tarea del registrador comente `%BFGCMD CMD=fteCreateLogger -h` insertando `/*` delante de la sentencia y asegurándose de que la columna uno esté en blanco.

Elimine los comentarios del segundo mandato y configure las sentencias. Por ejemplo:

```
%BFGCMD CMD=fteCreateLogger   +
-p MQPH     +
-loggerQMgr MQPH      +
-loggerType FILE        +
-fileLoggerMode circular    +
-fileSize 5MB +
-fileCount 5 +
-p MQPH +
-credentialsFile //'<MFTCredentialsDataSet(MemberName)>'
LOGGER
```

Compruebe la salida para ver si se ha procesado correctamente.

Consejo: Copie el nombre de vía de acceso del archivo `logger.properties` desde la salida del trabajo a un miembro del PDSE del agente.

Por ejemplo, cópielo en el miembro APATH

```
/u/user_ID/fte/wmqmft/mqft/config/MQPH/loggers/LOGGER/logger.properties
```

Esto es útil si necesita visualizar el archivo de propiedades.

Añada el directorio a este archivo:

```
/u/user_ID/fte/wmqmft/mqft/logs/MQPH/loggers/LOGGER/
```

Si está registrando en el archivo, los archivos de registro se almacenan en este directorio, por ejemplo, `LOGGER0-20140522123654897.log`.

Los archivos de rastreo están en el subdirectorio de registro, por ejemplo

```
/u/user_ID/fte/wmqmft/mqft/logs/MQPH/loggers/LOGGER/logs
```

Ahora puede iniciar la tarea de registro.

Creación de una tarea de registro al registrar en Db2

Cambie el nombre del miembro BFGLGCRS.

Este trabajo actualiza los archivos del directorio MFT y utiliza CSQUTIL para crear colas específicas del agente en el gestor de colas local.

Necesita saber:

Tabla 48. Variables de Db2

Nombre de Db2	**Ejemplo**
-dbName databaseName	Puede obtenerlo del valor de ubicación en el mensaje DSNL004I para el subsistema Db2
- dbDriver filePath	Por ejemplo `/db2/db2v10/jdbc/classes/db2jcc.jar`
-dbLib filePath	Por ejemplo `/db2/db2v10/jdbc/lib/libdb2jcct2zos_64.so`

Edite el archivo. El archivo original tiene el mandato `%BFGCMD CMD=fteCreateLogger -h`, el cual lista la sintaxis del mandato.

Elimine los comentarios del segundo mandato y configure las sentencias. Por ejemplo

```
%BFGCMD CMD=fteCreateLogger   +
-p MQPH     +
-loggerQMgr MQPH      +
-loggerType DATABASE      +
-dbType DB2           +
-dbName DSNDBCP         +
-dbDriver /db2/db2v10/jdbc/classes/db2jcc.jar  +
-dbLib /db2/db2v10/jdbc/lib/            +
-credentialsFile //'<MFTCredentialsDataSet(MemberName)>' +
LOGGER
```

Para crear la tarea del registrador comente `%BFGCMD CMD=fteCreateLogger -h` insertando `/*` delante de la sentencia y asegurándose de que la columna uno esté en blanco.

Someta el trabajo y compruebe la salida para ver si se ha procesado correctamente.

Consejo: Copie el nombre de vía de acceso del archivo `logger.properties` desde la salida del trabajo a un miembro del PDSE de los agentes.

Por ejemplo, cópielo en el miembro APATH:

```
/u/user_ID/fte/wmqmft/mqft/config/MQPH/loggers/LOGGER/logger.properties into member USS
```

Esto resulta útil si necesita visualizar el archivo de propiedades

Los archivos de rastreo están en el subdirectorio de registro, por ejemplo:

```
/u/user_ID/fte/wmqmft/mqft/logs/MQPH/loggers/LOGGER/logs
```

Creación de tablas de Db2

Debe crear las tablas de Db2. Las definiciones están en el archivo z/OS UNIX System Services `mqft/sql/ftelog_tables_zos.sql`.

Cree un miembro Db2 en su PDSE. Edite este miembro y utilice el mandato COPY en la línea de mandatos. Cópielo desde el archivo de definiciones de z/OS UNIX System Services.

Puesto que los requisitos específicos del sitio pueden variar mucho, este archivo solo especifica las estructuras básicas de las tablas, y un espacio de tabla donde se ubicarán.

El espacio de tabla se especifica, mediante el script SQL, para asegurarse de que se crea utilizando una agrupación de almacenamiento intermedio con un tamaño de página suficiente para contener las filas de tablas más grandes posibles. Tenga en cuenta que no se especifican atributos tales como ubicaciones LOB, etcétera.

Es posible que el administrador de su base de datos desee modificar una copia de este archivo para definir los atributos relacionados con el rendimiento.

Este archivo también presupone un nombre de esquema predeterminado FTEL0G, un nombre de espacio de tablas predeterminado FTEL0GTS y un nombre de base de datos FTEL0GDB. Puede cambiar estos nombres, si es necesario, para que coincidan con una base de datos existente y con las convenciones de denominación locales, siguiendo el proceso descrito en los comentarios que figuran al principio del archivo.

Importante: Utilice los recursos en línea, tales como **SPUFI** para ejecutar los mandatos, ya que existen comentarios en el archivo y los programas por lotes, tales como **DSNTINAD**, no aceptan comentarios.

Consulte Ejecución de SQL utilizando SPUFI para obtener más información. Además, CSQ45STB en SCSQPROC tiene JCL de ejemplo que puede personalizar para realizar los mandatos SELECT de Db2 .

Inicio de la tarea del registrador

Cambie el nombre, revise y someta el miembro BFGLGST. Debería recibir el mensaje BFGDB0023I: El registrador ha completado actividades de inicio y ahora se está ejecutando.

Operaciones del registrador

Para visualizar el estado del registrador, cambie el nombre, revise y envíe el miembro BFGLGSH.

Para detener el registrador, cambie el nombre, revise y envíe el miembro BFGLGSP.

z/OS Environment variables for MFT on z/OS

If you are running commands direct from the z/OS UNIX System Services (z/OS UNIX) environment, or your own JCL scripts, after customization and configuration you must set a number of environment variables before running the configuration and administration scripts provided by Managed File Transfer. You must set these variables for each user and in each environment that the scripts will be invoked from.

To avoid conflicts with other products, you can choose to create a `.wmqfterc` script in your home directory. The `.wmqfterc` script is then invoked by each of the Managed File Transfer scripts and you can use this script to provide custom environment settings for Managed File Transfer.

There is also one optional environment variable, BFG_WTO, that you can set to send messages to the operator log when running agents from JCL.

Table 49. Required z/OS environment variables

Environment variable	**Value**
BFG_JAVA_HOME	The location of your Java installation. For more information about the levels of Java supported, see System Requirements for IBM MQ.
BFG_DATA	The location of the data directory for Managed File Transfer for z/OS. This is the path to *DATA_DIR*.

Table 49. Required z/OS environment variables (continued)	
Environment variable	**Value**
STEPLIB	Must include the following IBM MQ data sets: • SCSQAUTH • SCSQANLE • SCSQLOAD If you want to run the database logger component on a z/OS system, STEPLIB must also include the following Db2 data sets in the order shown: • SDSNEXIT • SDSNLOD2 • SDSNLOAD

The following is an example `.profile` that correctly configures the environment variables for Managed File Transfer:

```
STEPLIB=MQM.V920.SCSQAUTH:MQM.V920.SCSQANLE:MQM.V920.SCSQLOAD
PATH=/u/fteuser/bin:/u/fteuser/J7.0/bin:/bin:/usr/bin:/u/fteuser/extras/bin:/bin:$PATH
BFG_JAVA_HOME=/u/fteuser/J7.0
BFG_DATA=/u/fteuser/DATA_DIR
export PATH STEPLIB BFG_JAVA_HOME BFG_DATA
```

Attention: The LIBPATH environment variable is no longer needed when calling **`fte*`** commands from a z/OS UNIX environment, and should be removed from any existing `.wmqfterc` script

Optionally, you can also set the following environment variables:

Table 50. Optional z/OS environment variable	
Environment variable	**Value**
BFG_WTO	One of the following values will enable BFG_WTO : • YES • ON • TRUE One of the following values will disable BFG_WTO. These values are not case sensitive. • NULL • NO • OFF • FALSE Enables z/OS logging. By default, this environment variable is disabled. Messages that are written to the agent event log are also written to the z/OS operator log facility, which allows easier access for automation products when you run an agent from JCL. The routing code is Programmer Information (11) and the descriptor code is Informational (12).

Table 50. Optional z/OS environment variable (continued)	
Environment variable	**Value**
BFG_GROUP_NAME	The `mqm` file group is typically associated with Managed File Transfer configuration data files and commands. Consequently, all users who are members of the `mqm` group can access, and make changes to the Managed File Transfer configuration. For more information, see File system permissions for MFT in IBM MQ. For a z/OS system, a file group is a z/OS UNIX filesystem entity, and the `mqm` file group is not necessarily defined. You can define an alternative, existing z/OS UNIX filesystem group for Managed File Transfer configuration data files by using the BFG_GROUP_NAME environment variable. For example, at the z/OS UNIX shell prompt: `export BFG_GROUP_NAME=FTEGB` which defines group FTEGB to be associated with any subsequently created configuration files for the current z/OS UNIX session. You can set BFG_GROUP_NAME to a blank value, or remove it.

z/OS Configuration properties files

A summary of the properties that are used in Managed File Transfer.

- The MFT coordination.properties file
- The MFT command.properties file
- The MFT agent.properties file
- Logger configuration properties file

z/OS Configuring MFT for the z/OS Automatic Restart Manager (ARM)

Managed File Transfer is an ARM enabled application.

Before you begin

For more information about enabling ARM, and defining ARM policies for your system, see Using the z/OS Automatic Restart Manager (ARM).

If you want to use the MFT DB Logger ability to automatically restart and reconnect to a Db2 database, ARM is the only supported restart manager available.

About this task

Using ARM, agents and loggers can be configured for restart by setting the agent/logger properties armELEMTYPE, and armELEMENT. Property armELEMTYPE defines the type of ARM element and property armELEMENT is the name of the element that ARM is to register:

- You can set the agent ELEMTYPE to SYSBFGAG, and armELEMENT can be set to correspond with the agent name.
- You can set the logger ELEMTYPE to SYSBFGLG, and armELEMENT can be set to correspond with the logger name.

Note: Agents and loggers that are configured for restart by ARM can only be successfully run from a batch job or a started task. Attempts to start the agent or logger from the z/OS UNIX System Services command line directly will fail with an ARM error reason code.

Example

The following example of a restart policy defines agent BFGFT7CAG1 as being dependant on queue manager FT7C:

```
RESTART_ORDER
   LEVEL(3)
   ELEMENT_TYPE(SYSBFGAG,SYSBFGLG)

RESTART_GROUP(GROUP7C)
   ELEMENT(SYSMQMGRFT7C)
   ELEMENT(BFGFT7CAG1)
       RESTART_ATTEMPTS(3,300)
```

z/OS Example: Creating JCL for Managed File Transfer agents on z/OS

Use this information to generate some JCL that can be used to create and start an agent on IBM MQ for z/OS.

Copy the sample library

Carry out the following procedure:

1. Make a copy of the library SCSQFCMD (see “Copy SCSQFCMD to create a JCL library” on page 771) by opening the library.

   The majority of the members, those that start with BFGX, BFGY, or BFGZ, are templates that you use to generate the customized JCL for the agent later on.

   The important member is BFGCOPY.
2. Open BFGCOPY and replace:

   ++supplied_library++
   with the name of the SCSQFCMD library that was installed as part of the product.

   ++service-library++
   with the name of the library that you want to use for your agent (the target library).
3. Submit the job and you have a new library that you can use.

Edit BFGCUSTM

Carry out the following procedure:

1. Open the new library so that you can edit the BFGCUSTM member (see “Editing member BFGCUSTM” on page 771)
2. Modify all of the parameters in the member that are enclosed within ++ characters, and replace them with the appropriate values. For example, change:

   ++mq_path++
   The path to the z/OS UNIX System Services (z/OS UNIX) Components directory. For example, `/mqm/V9R2M0`.

   Note: There are three instances of this variable to replace.

++bfg_data++
To be the z/OS UNIX directory where your IBM MQ Managed File Transfer for z/OS configuration is to be stored.

++service_type++
To the word AGENT

++agent_name++
To be the name of your agent

Notes:

1. Some of the entries, such as ++options++ required for the CLEAN_AGENT_PROPS, are not needed and so you should remove these.
2. See "Antes de empezar a configurar MFT para z/OS" on page 766 for a complete list of all of the parameters in the BFGCUSTM member, along with a description of what values they should have.

Submit the BFGCUSTM JCL

Carry out the following procedure:

1. Submit the job.
2. Exit the library in ISPF.

   This is necessary because the BFGCUSTM job is updating the library, and cannot do that while the library is open.
3. When the job completes look at the joblog.

   You will see a number of messages, indicating that new members have been created within the library.

   Each of these members contains JCL that can be used to perform specific tasks for your agent. See "z/OS agent and logger command JCL scripts" on page 779 for a list of these members, along with the IBM MQ Managed File Transfer commands that they correspond to.

Submit BFGAGCR to create the agent

The new member BFGAGCR contains some JCL that creates an agent by invoking the **fteCreateAgent** command.

Carry out the following procedure:

1. Open up member BFGAGCR.

   You should see that BFGAGCR has been populated with the name of your:

   - Agent
   - Agent queue manager
   - The coordination queue manager for the MFT topology
2. Submit member BFGAGCR.

   When the member runs, it:

   - Creates the required configuration files for your agent.
   - Connects to the agent queue manager, and creates the system queues that the agent needs, using CSQUTIL.
   - Registers the agent with the coordination queue manager.

Start the agent by submitting BFGAGST

Carry out the following procedure:

1. Submit the BFGAGST member. See using the agent for various commands that show you that the agent is working correctly.

2. When the job completes, check the joblog contains the following messages:

```
BFGAG0058I: The agent has successfully initialized.
BFGAG0059I: The agent has been successfully started.
```

   which means that your agent is up, running, and ready to perform managed transfers.

z/OS Moving an MFT agent to a new z/OS LPAR

It is sometimes necessary to move an IBM MQ Managed File Transfer for z/OS agent from one LPAR to another, while keeping the agent in the same IBM MQ Managed File Transfer topology with the same coordination and command queue managers. The steps needed to do this depend on how the agent being migrated was originally created.

About this task

Move your IBM MQ Managed File Transfer for z/OS agent in one of the following ways:

- If the agent was originally created using a customized version of the SCSQFCMD library, use the library to recreate it on a new LPAR.
- If the agent was originally created by running z/OS UNIX System Services (z/OS UNIX) commands, use the commands to recreate it on a new LPAR.

Note:

Scheduled transfers and transfer templates are stored on the coordination queue manager for an IBM MQ Managed File Transfer topology. This task assumes that the coordination queue manager is not part of the movement work. In this case, any scheduled transfers and transfer templates associated with the agent being moved remain on the existing coordination queue manager after the move is completed.

Procedure

- Move an agent created using a customized version of the SCSQFCMD library.

  If the agent was created using a customized version of the SCSQFCMD library, you can use that library to recreate the IBM MQ Managed File Transfer for z/OS environment, and the agent configuration on the new LPAR. To do this, complete the following steps:

  1. Copy the customized version of the library from the original LPAR to the new LPAR.
  2. Edit the BFGCUSTM member in the customized version of the library on the new LPAR, and make sure that the parameter values are still valid.
  3. Run the BFGCUSTM member on the new LPAR, to create all of the JCL needed to configure the environment and create the agent.
  4. Run the BFGCFCR member to define the coordination queue manager to be used by the agent on the new LPAR, and create the directory structure needed to store the IBM MQ Managed File Transfer configuration.
  5. Next, run the BFGCMCR member, to define the command queue manager to be used by the agent on the new LPAR.
  6. Run the BFGAGCR member to recreate the agent and its configuration.
  7. Ensure that the system queues used by the agent exist on the queue manager for that agent.

  If the agent being moved has resource monitors associated with it, you need to recreate the monitors on the new agent. To do this, complete the following steps:

  1. On the original LPAR, run the BFGMNLI member to export the definitions for the resource monitor associated with the original agent to XML files.
  2. Copy the XML files containing the resource monitor definitions to the new LPAR.

3. Use the BFGMNCRS member in the SCSQFCMD library on the new LPAR to import the resource monitor definitions stored in the XML files. This results in the monitors being created on the new agent.

- Move an agent created by running commands in z/OS UNIX.

  If the agent was originally created by running z/OS UNIX commands, you can use commands to recreate the agent on a new LPAR. To do this, complete the following steps:

  1. Run the fteSetupCoordination command on the new LPAR, to define the coordination queue manager to be used by the agent, and create the directory structure needed to store the IBM MQ Managed File Transfer configuration.
  2. Run the fteSetupCommands command to define the command queue manager to be used by the agent on the new LPAR.
  3. Run the fteCreateAgent command to recreate the agent and its configuration.
  4. Ensure that the system queues used by the agent exist on the queue manager for that agent.

  If the agent being moved has resource monitors associated with it, you need to recreate the monitors on the new agent. To do this, complete the following steps:

  1. On the original LPAR, run the fteListMonitors command, specifying the **-ox** parameter, to export the definitions for the resource monitor, associated with the original agent, to XML files.
  2. Copy the XML files containing the resource monitor definitions to the new LPAR.
  3. Run the fteCreateMonitor command on the new LPAR, specifying the **-ix** parameter, to import the resource monitor definitions stored in the XML files. This results in the monitors being created on the new agent.

z/OS Planning your MFT infrastructure with IBM MQ for z/OS queue sharing groups

You need to consider the following, if you are using IBM MQ Managed File Transfer (MFT) , when one or more of the agents, command or coordination queue managers are part of an IBM MQ for z/OS queue sharing group.

See MFT topology overview for a description of agents, command queue managers, and coordination queue managers.

Agent queue managers

Normally an MFT agent connects to a single agent queue manager, and uses local queues that are only accessible by that queue manager. The agent is informed which queue manager to connect to, by providing it with the queue manager name when the agent is first created.

With IBM MQ for z/OS, it is possible to create the agent and replace the queue manager name with the name of a queue sharing group (QSG). This means that the agent can connect to any available queue manager in the QSG to perform file transfers. Should there be a failure of the queue manager that the agent is currently connected to, the agent detects the failure and reconnects to an alternative queue manager in the QSG.

Connecting an agent to a QSG in combination with the highly available agent support provided allows very robust MFT topologies to be created. See “Agentes de alta disponibilidad en Managed File Transfer” on page 813.

For example, in the following figure *Agent1* has been created so that its agent queue manager is a QSG consisting of two queue managers *QM1* and *QM2*. The agent queues have been defined as shared queues, stored in the coupling facility.

This means that the agent can run on either *LPAR 1* or *LPAR 2* and connect to either *QM1* or *QM2*. The files and data sets that the agent reads from, or writes to, are shared, meaning they can be accessed from either LPAR.

In addition, the agent has been configured to be a highly available agent. In the diagram the agent is active in *LPAR 1* and a standby instance of the agent is running in *LPAR 2*.

This topology provides high resilience. Should either the agent running on *LPAR 1* fail, or queue manager *QM1* fail, or *LPAR 1* fail the standby instance of the agent on *LPAR 2* can take over and carry on processing file transfers from the point of failure.

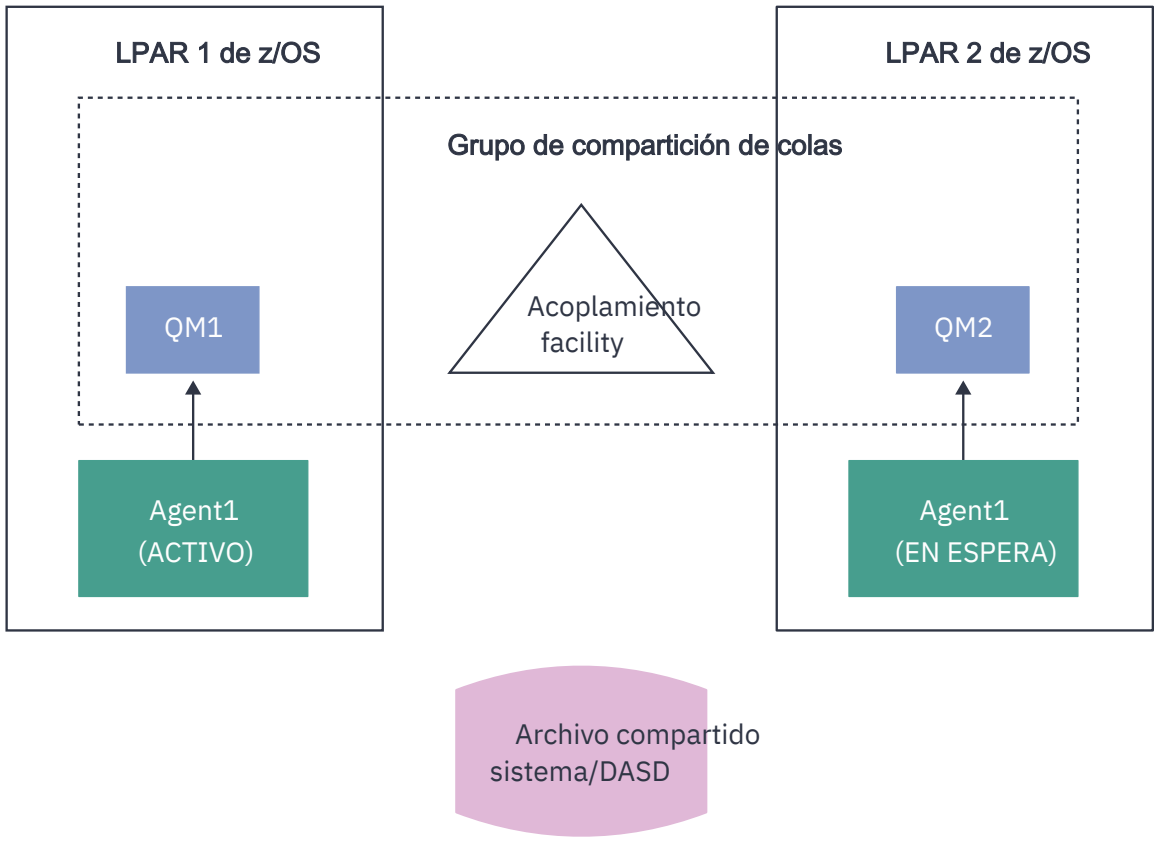


Figure 95. Highly available MFT agent using a queue sharing group

Creating an agent that uses a QSG as the agent queue manager

You create an agent using the fteCreateAgent command. When doing this, the name of the queue sharing group is provided for the agent queue manager. For example:

```
fteCreateAgent -agentName Agent1 -agentQMgr QSG1
```

This creates an agent called *Agent1* which uses any queue manager that is a member of QSG *QSG1* as its agent queue manager. In this configuration the agent connects to the agent queue manager using a cross memory (bindings mode) connection which means that the agent and the queue manager must be on the same LPAR. This is exactly like the example shown in figure 1 above.

When you run the **fteCreateAgent** command it generates a set of MQSC commands to create the necessary queues on the agent queue manager.

When the agent queue manager is a QSG, this set of commands needs to be modified so that each queue is created as a shared queue. That is, each queue needs to be created with QSGDISP(SHARED) and an appropriate coupling facility structure provided by the CFSTRUCT attribute.

The following example shows you how to change the MQSC command for creating the SYSTEM.FTE.COMMAND.AGENT1 queue as a shared queue. The changes to the defaults are in bold text.

Important: You need to make similar changes to all the other queues that the agent uses.

```
DEFINE QLOCAL(SYSTEM.FTE.COMMAND.AGENT1) +
 QSGDISP(SHARED) +
 CFSTRUCT(MFTSTRUCT) +
 DEFPRTY(0) +
 DEFSOPT(SHARED) +
 GET(ENABLED) +
 INDXTYPE(CORRELID) +
 MAXDEPTH(5000) +
 MAXMSGL(4194304) +
 MSGDLVSQ(PRIORITY) +
 PUT(ENABLED) +
 RETINTVL(999999999) +
 SHARE +
 NOTRIGGER +
 USAGE(NORMAL) +
 REPLACE
```

Creating an agent that uses a QSG as the agent queue manager and connects as a client

Agents can connect to their agent queue manager using a client channel. You can use this approach to allow the agent to run on distributed platforms while connecting to a QSG. If all queue managers in the QSG are licensed for IBM MQ Advanced for z/OS Value Unit Edition, then the agent can also connect to them from a z/OS LPAR that does not have a local queue manager.

This topology is shown in the following figure and allows the agent to take advantage of the resiliency of QSGs. If the queue manager in the QSG that the agent is currently connected to fails, then the agent automatically reconnects to a different member of the QSG and carries on processing.

The sysplex distributor is used to spread the connections from the agent across the available queue managers in the QSG.

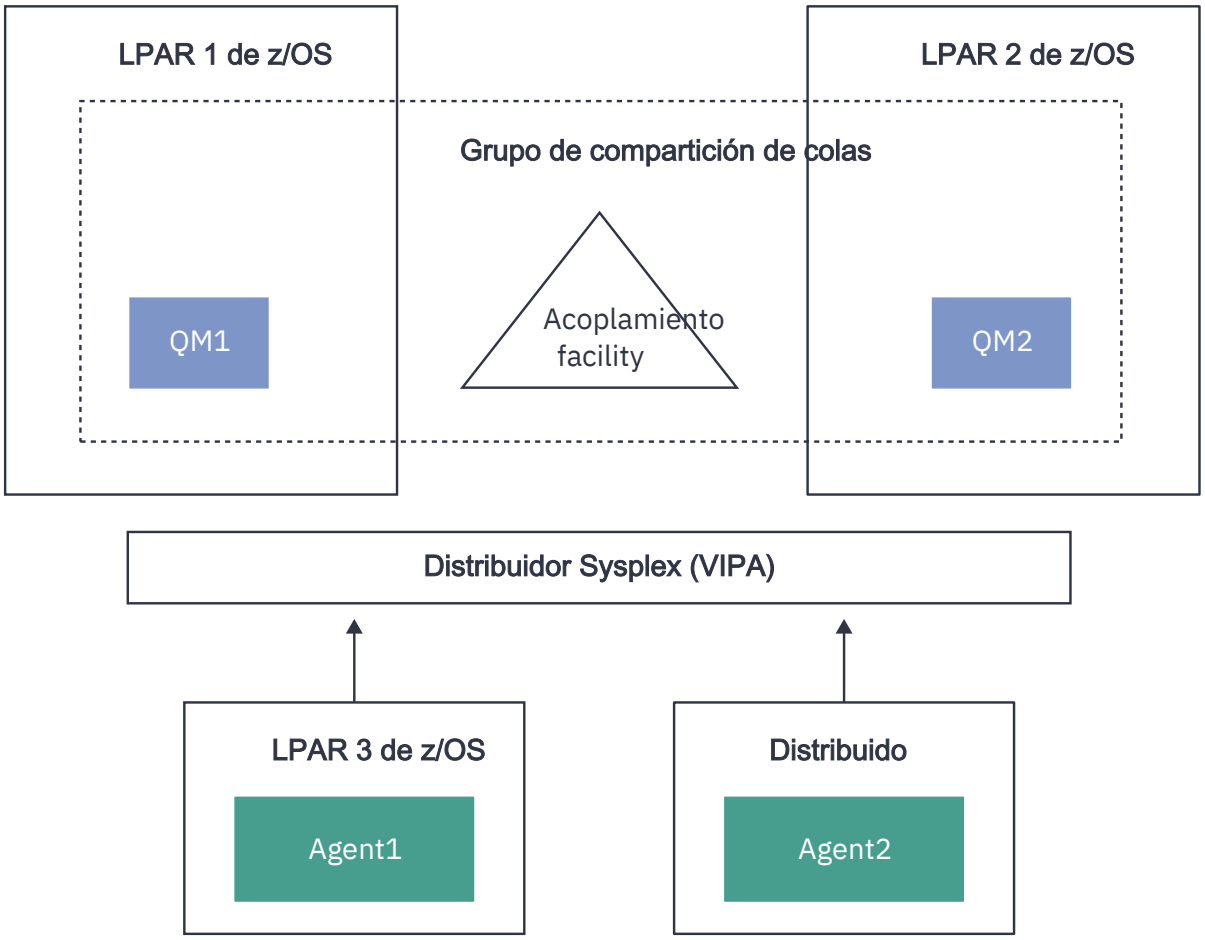


Figure 96. MFT agents connecting to a queue sharing group as a client

In order to make use of this topology, the queue managers in the QSG must each have a server connection channel defined for use by the agent. See "Connecting a client to a queue sharing group" on page 65 for information on how to do this.

When creating the agent the queue managers need to be configured, so that they can use the channel that is defined to the QSG, and access it through the sysplex distributor. For example:

```
fteCreateAgent -agentName Agent1 -agentQMgr QSG1 -agentQMgrHost vipaAddress
-agentQMgrPort sharedPort -agentQMgrChannel CHANNEL1
```

As previously stated the MQSC commands generated by running the **fteCreateAgent** command must be adjusted to specify QSGDISP(SHARED) and an appropriate coupling facility structure in the CFSTRUCT attribute.

Command queue managers

The MFT command queue manager can be part of a QSG. However, the name of a QSG cannot be used when specifying a command queue manager; you must use a specific queue manager name.

Coordination queue managers

The MFT coordination queue manager can be part of a QSG. However, as with a command queue manager, the name of a QSG cannot be used when specifying a coordination queue manager; you must use a specific queue manager name.

Commands connecting to a QSG

MFT provides a number of commands to managed agents, transfers and agent, command, or coordination queue managers. You can use only those commands that connect to an agent queue manager, if the queue manager is in a QSG.

Following is a list of the commands that connect to the agent queue manager:

- **fteCleanAgent**
- **fteCreateAgent**
- **fteCreateBridgeAgent**
- **fteCreateCDAgent**
- **fteDeleteAgent**

Note that you must provide the name of the queue manager when running other MFT commands.

Using Managed File Transfer for z/OS with the JZOS Java launcher

You can apply the instructions in this topic as an alternative method of using Managed File Transfer in your enterprise, on your IBM MQ for z/OS system.

Overview

Managed File Transfer for z/OS (MFT) uses the standard z/OS installation procedure. An alternative way of running MFT commands is to use JCL and the JZOS Java Launcher.

See JZOS Batch Launcher and Toolkit for further details.

If your JCL fails to process correctly, see Common MFT problems with JZOS.

Example JCL

```
//JOHNDOEA JOB 1,MSGCLASS=H
// JCLLIB ORDER=(SCEN.MFT.JCL)   (1)
// INCLUDE MEMBER=BFGJCL8       (2)
//     DD * (2A)
. ${BFG_PROD}/bin/fteBatch createAgent  (3)
export IBM_JAVA_OPTIONS="${BFG_JAVA_OPTIONS} ${BFG_LANG}"  (4)
export JZOS_MAIN_ARGS="${BFG_MAIN_ARGS}" (4)
//MAINARGS DD *
-agentName MYAGENT  (5)
-f
-agentQMgr MQPD
-p MQPD
/*
```

where:

- (1) Is the location of included JCL statements
- (2) Include the specified JCL member from the location in 1)
- (2A) This extends the //STDENV - see below
- (3) This is the command to be executed, without the leading fte prefix
- (4) These lines are required, they set up information for JZOS
- (5) The parameters to the command
- The BFGJCL8 member (you can select your own name) invokes JZOS. This member has the STEPLIB and other JCL needed to run MFT.

Other JCL you need to include

You should include JCL for the IBM MQ for z/OS libraries, and if you are using the Db2 logger, the Db2 libraries.

For example:

```
//WMQFTE EXEC PGM=JVMLDM86,REGION=0M PARM='+T'  (1)
//STEPLIB  DD DSN=SYS1.SIEALNKE,DISP=SHR        (2)
//* MQ libraries
//   DD DSN=MQM.V920.SCSQAUTH,DISP=SHR MQ Bindings
//   DD DSN=MQM.V920.SCSQANLE,DISP=SHR MQ Bindings
//   DD DSN=MQM.V920.SCSQLOAD,DISP=SHR MQ Bindings

//* DB2 libraries
//   DD DISP=SHR,DSN=SYS2.DB2.V12.SDSNEXIT.DBCP
//   DD DISP=SHR,DSN=SYS2.DB2.V12.SDSNLOAD
//   DD DISP=SHR,DSN=SYS2.DB2.V12.SDSNLOD2
//SYSOUT  DD SYSOUT=H
//SYSPRINT DD SYSOUT=H
//STDOUT  DD SYSOUT=H
//STDERR  DD SYSOUT=H

//STDENV  DD DSN=SCEN.MFT.JCL(BFGZENV8),DISP=SHR  (3)
```

where:

- (1) Is the name of the JZOS program. Look in SYS1.SIEALNKE for the version on your system. Add ,PARM='+T' to give additional diagnostics.
- (2) This is the data set with the JZOS program.
- (3) This is the member name of a shell script. It defines parameters needed by MFT. See “Shell script to define MFT” on page 795.

  It can be any data set and member. It needs to be last in the file because the JCL job extends this. See 2A in “Example JCL” on page 794.

Shell script to define MFT

In the “Other JCL you need to include” on page 794 example, the member BFGZENV8 is used. This is based on the JZOS profile.

You need to know:

- The location where Java is installed
- The location of the IBM MQ for z/OS Java libraries and the MFT libraries.
- A user ID needs to be in a specific group to be considered as an IBM MQ for z/OS administrator. You need the name of this group
- If you are not using English for the messages, you need to know which language to specify.

Example file

```
# This is a shell script that configures
# any environment variables for the Java JVM.
# Variables must be exported to be seen by the launcher.
# Use PARM='+T' and set -x to debug environment script problems
set -x
# . /etc/profile
#
# Java configuration (including MQ Java interface)
#
export _BPXK_AUTOCVT="ON"
export JAVA_HOME="/java/java71_bit64_sr3_fp30/J7.1_64/"
export PATH="/bin:${JAVA_HOME}/bin/classic/"
LIBPATH="/lib:/usr/lib:${JAVA_HOME}/bin"
LIBPATH="$LIBPATH:${JAVA_HOME}/bin/classic"
LIBPATH=$LIBPATH:"/mqm/V9R2M0/java/lib/"
export LIBPATH

export BFG_JAVA_HOME="${JAVA_HOME}"
export BFG_WTO="YES"
export BFG_GROUP_NAME=MQADM
export BFG_PROD="/mqm/V9R2M0/mqft"
export BFG_CONFIG="/u/johndoe/fteconfig"
```

```
# export BFG_LANG=" -Duser.language=de "
export BFG_LANG=" "

--------------------
```

where:

export _BPXK_AUTOCVT="ON"
Is required for Unicode conversion

export JAVA_HOME="/java/java71_bit64/J7.1_64/"
Is the location of the Java directory. Specify the name of the path for Java. This directory contains bin and other directories.

export PATH="/bin:${JAVA_HOME}/bin/classic/"
Sets up the path statement for Java executable statements

LIBPATH="/lib:/usr/lib:${JAVA_HOME}/bin"
Sets up the library path for the Java executable statements

LIBPATH="$LIBPATH:${JAVA_HOME}/bin/classic"
Adds more Java libraries to the LIBPATH statement.

LIBPATH=$LIBPATH:"/mqm/V9R2M0/java/lib/"
Adds IBM MQ for z/OS libraries in the library path. Specify the name of your IBM MQ for z/OS libraries in z/OS UNIX System Services.

export LIBPATH
Makes the LIBPATH available to JZOS

export BFG_JAVA_HOME="${JAVA_HOME}"
Sets the BFG_JAVA_HOME to the value of JAVA_HOME specified above

export BFG_WTO="YES"
Setting BFG_WTO to YES causes messages to be displayed on the joblog using WTO

export BFG_GROUP_NAME=MQADM
User IDs, which are a member of the specified group, are considered IBM MQ for z/OS administrators

export BFG_PROD="/mqm/V9R2M0/mqft"
Is the path where the MFT code is located

export BFG_DATA="/u/johndoe/fteconfig"
Is where the MFT configuration information is stored

 # export BFG_LANG=" -Duser.language=de "
Is a commented out statement to define the language as German

export BFG_LANG=" "
Specifies the language as the default, English.

The contents of the MFT product in `/lib/messages/BFGNVMessages_*.properties` lists the languages available. The default is to leave the value blank, which means that English is used.

Related tasks

“Configuring Managed File Transfer for z/OS” on page 764
Managed File Transfer for z/OS requires customization to enable the component to operate correctly.

Planning for Managed File Transfer

IBM i Configuración de MFT en IBM i

Para empezar a utilizar Managed File Transfer después de haberlo instalado, debe realizar algunos pasos de configuración para el gestor de colas de coordinación y el agente.

Acerca de esta tarea

Después de haber realizado la instalación, debe ejecutar los scripts de configuración proporcionados por Managed File Transfer para nuevos gestores de colas de coordinación y nuevos agentes antes de poder

utilizar los gestores de colas de coordinación y los agentes para transferir archivos. A continuación, debe iniciar los agentes que ha creado.

Procedimiento

1. Para todos los nuevos gestores de colas de coordinación: ejecute los mandatos MQSC en el archivo *coordination_qmgr_name*.mqsc en el gestor de colas de coordinación. Si el gestor de colas de coordinación no se encuentra en el mismo sistema que el agente, copie el archivo de script MQSC en el sistema en el que se encuentra el gestor de colas y, a continuación, ejecute el script.
   a) Desde una línea de mandatos de IBM i, inicie qshell utilizando el mandato siguiente: `CALL QSHELL`
   b) Cambie al directorio siguiente: `/QIBM/UserData/mqm/mqft/config/`*coordination_qmgr_name*
   c) Emita el siguiente mandato, sustituyendo *nombre_gestcolas_coordinación* por el nombre de gestor de colas:

   ```
   /QSYS.LIB/QMQM.LIB/RUNMQSC.PGM coordination_qmgr_name < coordination_qmgr_name.mqsc
   ```

   En su lugar puede configurar el gestor de colas de coordinación manualmente. Para obtener más información, consulte "Configuración del gestor de colas de coordinación para MFT" en la página 801.
2. Para todos los agentes nuevos: ejecute los mandatos MQSC en el archivo *agent_name*_create.mqsc en el gestor de colas del agente.
   Si el gestor de colas del agente no se encuentra en el mismo sistema que el agente, copie el archivo de script MQSC en el sistema en el que se encuentra el gestor de colas y, a continuación, ejecute el script.
   a) Desde una línea de mandatos de IBM i, inicie qshell utilizando el mandato siguiente: `CALL QSHELL`
   b) Cambie al directorio siguiente: `/QIBM/UserData/mqm/mqft/config/`*agent_qmgr_name*`/agents`
   c) Emita el siguiente mandato, sustituyendo *nombre_gestcolas_agente* por el nombre del gestor de colas del agente y sustituyendo *nombre_agente* por el nombre del agente.

   ```
   /QSYS.LIB/QMQM.LIB/RUNMQSC.PGM agent_qmgr_name < agent_name_create.mqsc
   ```

   En su lugar, puede configurar el gestor de colas del agente manualmente. Para obtener más información, consulte "Configuración de gestores de colas de agente de MFT" en la página 808.
3. Si todavía no ha iniciado el subsistema QMFT como parte de la instalación, desde la línea de mandatos de IBM i , inicie el subsistema QMFT utilizando el mandato siguiente: `STRSBS SBSD(QMQMMFT/QMFT)`, o `STRSBS QMQMMFT/QMFT`
4. Inicie los nuevos agentes utilizando el mandato **fteStartAgent**.
   a) Desde una línea de mandatos de IBM i, inicie qshell utilizando el mandato siguiente: `CALL QSHELL`
   b) Cambie al directorio siguiente: `/QIBM/ProdData/mqm/bin`
   c) Emita el siguiente mandato, sustituyendo AGENT por el nombre del agente:

   ```
   ./fteStartAgent AGENT
   ```

Qué hacer a continuación

Para limitar las áreas del sistema de archivos a las que un agente puede acceder, se recomienda configurar recintos de seguridad. Esta característica se describe en Trabajar con recintos de seguridad del agente MFT.

Conceptos relacionados

"Configuración de MFT cuando se utiliza por primera vez" en la página 798

Debe realizar algunas tareas de configuración para agentes y gestores de colas de Managed File Transfer una sola vez, la primera vez que desea utilizarlos.

Configuración de MFT cuando se utiliza por primera vez

Debe realizar algunas tareas de configuración para agentes y gestores de colas de Managed File Transfer una sola vez, la primera vez que desea utilizarlos.

Conceptos relacionados

"conexión a IBM MQ" en la página 798
Todas las comunicaciones de red con los gestores de colas de IBM MQ, incluyendo la comunicación relacionada con Managed File Transfer, implican canales de IBM MQ. Un canal de IBM MQ representa un extremo de un enlace de red. Los canales se clasifican como canales de mensajes o canales MQI.

"Configuración de un gestor de colas multiinstancia para que funcione con MFT" en la página 805
IBM WebSphere MQ 7.0.1 y posterior es acepta la creación de gestores de colas multiinstancia. Un gestor de colas multiinstancia se reinicia automáticamente en un servidor en espera. Managed File Transfer soporta la conexión a gestores de colas de agente multiinstancia, a un gestor de colas de coordinación multiinstancia y a un gestor de colas de mandatos multiinstancia.

Tareas relacionadas

"Configuración de gestores de colas de red de MFT" en la página 800
Si la red de Managed File Transfer incluye más de un gestor de colas de IBM MQ, estos gestores de colas de IBM MQ deben poder comunicarse de forma remota entre sí.

"Configuración de gestores de colas de agente de MFT" en la página 808
Después de la instalación, ejecute el script *agent_name*_create.mqsc en el directorio *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name*/agents/*agent_name* para realizar la configuración necesaria para el gestor de colas del agente. Sin embargo, si desea realizar esta configuración manualmente, complete estos pasos en el gestor de colas de agente.

"Configuración del gestor de colas de coordinación para MFT" en la página 801
Después de ejecutar el mandato **fteSetupCoordination** , ejecute el script *coordination_qmgr_name*.mqsc en el directorio *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name* para realizar la configuración necesaria para el gestor de colas de coordinación. Sin embargo, si desea realizar esta configuración manualmente, realice los pasos siguientes en el gestor de colas de coordinación.

"Creating an MFT Agent or Logger command data set" en la página 763
You can create a PDSE data set of commands from the Managed File Transfer command template data set for a specific Managed File Transfer Agent or Managed File Transfer Logger for a specific coordination.

"Updating an existing MFT Agent or Logger command data set on z/OS" en la página 775
You can update an Managed File Transfer command PDSE library data set that is created from the Managed File Transfer command template data set.

Referencia relacionada

Valores de cola del agente MFT

Colas de sistema de MFT y el tema de sistema

"Conservación de mensajes de registro de MFT" en la página 807
Managed File Transfer envía información de progreso y de registro de la transferencia de archivos al gestor de colas de coordinación. El gestor de colas de coordinación publica esta información en las suscripciones coincidentes al tema SYSTEM.FTE. Si no existe ninguna suscripción, esta información no se conserva.

conexión a IBM MQ

Todas las comunicaciones de red con los gestores de colas de IBM MQ, incluyendo la comunicación relacionada con Managed File Transfer, implican canales de IBM MQ. Un canal de IBM MQ representa un extremo de un enlace de red. Los canales se clasifican como canales de mensajes o canales MQI.

Managed File Transfer y los canales

Managed File Transfer utiliza canales MQI para conectar los agentes en modalidad de cliente a sus gestores de colas de agente, y también para conectar aplicaciones de mandatos (por ejemplo **fteCreateTransfer**) a sus gestores de colas de mandatos y de coordinación. En la configuración predeterminada, estas conexiones se efectúan utilizando un canal SVRCONN denominado SYSTEM.DEF.SVRCONN, que existe de forma predeterminada en todos los gestores de colas. Debido a estos valores predeterminados, no es necesario alterar ningún canal MQI para una instalación básica de Managed File Transfer.

Hay seis tipos de puntos finales de canal de mensajes, pero este tema sólo describe los pares emisor-receptor. Consulte Componentes de gestión de colas distribuidas para obtener información sobre otras combinaciones de canales.

Vías de acceso de mensajes necesarias

Los mensajes de IBM MQ sólo pueden desplazarse por canales de mensajes, por lo que debe garantizarse que los canales estén disponibles para todas las vías de acceso de mensajes que Managed File Transfer necesita. Estas vías de acceso no tienen que ser directas; los mensajes pueden desplazarse a través de gestores de colas intermedios si es necesario. Este tema sólo hace referencia a las comunicaciones punto a punto. Consulte Cómo llegar al gestor de colas remoto para obtener más información sobre estas opciones.

Las vías de acceso de comunicación utilizadas por Managed File Transfer son las siguientes:

Agente a agente
: Siempre que hay dos agentes que transfieren archivos entre ellos, es necesaria la comunicación bidireccional entre sus gestores de colas asociados. Puesto que esta vía de acceso transporta datos en masa, es aconsejable que sea lo más corta, rápida y económica posible en función de sus necesidades.

Agente a coordinación
: Los mensajes de registro de los agentes que participan en una transferencia deben ser capaces de llegar al gestor de colas de coordinación.

Mandato a agente
: Todos los gestores de colas a los que se conecten las aplicaciones de mandatos o IBM MQ Explorer (utilizando el gestor de colas de mandatos) deben ser capaces de enviar mensajes a los gestores de colas de los agentes que dichas aplicaciones de mandatos suelen controlar. Para permitir que los mensajes de respuesta se muestren en los mandatos, utilice una conexión bidireccional.

Para obtener más información, consulte *Verificación de una instalación de IBM MQ* para la plataforma, o plataformas, que utiliza su empresa.

Conceptos relacionados

“Configuración de un gestor de colas multiinstancia para que funcione con MFT” en la página 805
IBM WebSphere MQ 7.0.1 y posterior es acepta la creación de gestores de colas multiinstancia. Un gestor de colas multiinstancia se reinicia automáticamente en un servidor en espera. Managed File Transfer soporta la conexión a gestores de colas de agente multiinstancia, a un gestor de colas de coordinación multiinstancia y a un gestor de colas de mandatos multiinstancia.

Tareas relacionadas

“Configuración de gestores de colas de red de MFT” en la página 800
Si la red de Managed File Transfer incluye más de un gestor de colas de IBM MQ, estos gestores de colas de IBM MQ deben poder comunicarse de forma remota entre sí.

“Configuración del gestor de colas de coordinación para MFT” en la página 801
Después de ejecutar el mandato **fteSetupCoordination** , ejecute el script *coordination_qmgr_name*.mqsc en el directorio *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name* para realizar la configuración necesaria para el gestor de colas de coordinación. Sin embargo, si desea realizar esta configuración manualmente, realice los pasos siguientes en el gestor de colas de coordinación.

Configuración de gestores de colas de red de MFT

Si la red de Managed File Transfer incluye más de un gestor de colas de IBM MQ, estos gestores de colas de IBM MQ deben poder comunicarse de forma remota entre sí.

Acerca de esta tarea

Existen dos maneras de configurar los gestores de colas para que se comuniquen entre sí:

- Configurando un clúster de gestor de colas de IBM MQ.

  Si desea más información sobre clústeres de gestores de colas IBM MQ y cómo configurarlos, consulte “Configuración de un clúster de gestores de colas” en la página 309.
- Configurando canales entre los gestores de colas, que se describe a continuación:

Configuración de canales entre gestores de colas

Configure los siguientes canales de mensaje entre los gestores de colas:

- Desde el gestor de colas del agente hasta el gestor de colas de coordinación
- Desde el gestor de colas de mandatos hasta el gestor de colas de agente.
- Desde el gestor de colas del agente hasta el gestor de colas de mandatos (para permitir que los mensajes de respuesta se muestren en los mandatos).
- Desde el gestor de colas de mandatos hasta el gestor de colas de coordinación
- Desde el gestor de colas del agente hasta cualquier otro gestor de colas de agente de la red de Managed File Transfer

Si necesita más información acerca de cómo establecer esta comunicación, empiece con esta información: Administración de objetos IBM MQ remotos utilizando MQSC.

Algunos de los pasos de ejemplo que se recomiendan son los siguientes:

Procedimiento

1. Cree una cola de transmisión en el gestor de colas de IBM MQ con el mismo nombre que el gestor de colas de coordinación.

   Puede utilizar el siguiente mandato MQSC:

   ```
   DEFINE QLOCAL(coordination-qmgr-name) USAGE(XMITQ)
   ```

2. En el gestor de colas de IBM MQ, cree un canal emisor al gestor de colas de coordinación de Managed File Transfer.

   El nombre de la cola de transmisión creado en el paso anterior es un parámetro necesario para este canal.

   Para agentes en Managed File Transfer para IBM MQ, los mensajes se publican con un formato en blanco.

   Puede utilizar el siguiente mandato MQSC:

   ```
   DEFINE CHANNEL(channel-name) CHLTYPE(SDR) CONNAME('coordination-qmgr-host(coordination-qmgr-port)')
    XMITQ(coordination-qmgr-name) CONVERT(NO)
   ```

   Nota: Establezca CONVERT(NO), sólo si es necesario.

3. En el gestor de colas de Managed File Transfer, cree un canal receptor al gestor de colas de IBM MQ. Asigne a este canal receptor el mismo nombre que el canal emisor en el gestor de colas de IBM MQ.

   Puede utilizar el siguiente mandato MQSC:

   ```
   DEFINE CHANNEL(channel-name) CHLTYPE(RCVR)
   ```

Qué hacer a continuación

A continuación, siga los pasos de configuración para el gestor de colas de coordinación: "Configuración del gestor de colas de coordinación para MFT" en la página 801.

Conceptos relacionados

"conexión a IBM MQ" en la página 798
Todas las comunicaciones de red con los gestores de colas de IBM MQ, incluyendo la comunicación relacionada con Managed File Transfer, implican canales de IBM MQ. Un canal de IBM MQ representa un extremo de un enlace de red. Los canales se clasifican como canales de mensajes o canales MQI.

"Configuración de un gestor de colas multiinstancia para que funcione con MFT" en la página 805
IBM WebSphere MQ 7.0.1 y posterior es acepta la creación de gestores de colas multiinstancia. Un gestor de colas multiinstancia se reinicia automáticamente en un servidor en espera. Managed File Transfer soporta la conexión a gestores de colas de agente multiinstancia, a un gestor de colas de coordinación multiinstancia y a un gestor de colas de mandatos multiinstancia.

Tareas relacionadas

"Configuración del gestor de colas de coordinación para MFT" en la página 801
Después de ejecutar el mandato **fteSetupCoordination** , ejecute el script *coordination_qmgr_name*.mqsc en el directorio *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name* para realizar la configuración necesaria para el gestor de colas de coordinación. Sin embargo, si desea realizar esta configuración manualmente, realice los pasos siguientes en el gestor de colas de coordinación.

Configuración del gestor de colas de coordinación para MFT

Después de ejecutar el mandato **fteSetupCoordination** , ejecute el script *coordination_qmgr_name*.mqsc en el directorio *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name* para realizar la configuración necesaria para el gestor de colas de coordinación. Sin embargo, si desea realizar esta configuración manualmente, realice los pasos siguientes en el gestor de colas de coordinación.

Acerca de esta tarea

Procedimiento

1. Cree una cola local denominada SYSTEM.FTE.
2. Añada la cola SYSTEM.FTE a la lista de nombres SYSTEM.QPUBSUB.QUEUE.NAMELIST.
3. Cree un tema denominado SYSTEM.FTE con una serie de tema de SYSTEM.FTE.
4. Compruebe que los atributos de entrega de mensajes no persistente (NPMSGDLV) y de entrega de mensajes persistente (PMSGDLV) del tema SYSTEM.FTE están establecidos en ALLAVAIL.
5. Asegúrese de que el atributo de modalidad de publicación/suscripción (PSMODE) del gestor de colas de coordinación está establecido en ENABLED.

Qué hacer a continuación

Si ejecuta el mandato `strmqm -c` en un gestor de colas que se ha configurado como gestor de colas de coordinación, el mandato suprime el cambio realizado en el paso 2 (añadiendo SYSTEM.FTE a SYSTEM.QPUBSUB.QUEUE.NAMELIST lista de nombres). Esto se debe a que `strmqm -c` vuelve a crear los objetos de IBM MQ predeterminados e invierte los cambios de Managed File Transfer. Por lo tanto, si ha iniciado el gestor de colas con `strmqm -c`, realice uno de los pasos siguientes:

- Vuelva a ejecutar el script *coordination_qmgr_name*.mqsc en el gestor de colas.
- Repita el paso 2.

Conceptos relacionados

"conexión a IBM MQ" en la página 798

Todas las comunicaciones de red con los gestores de colas de IBM MQ, incluyendo la comunicación relacionada con Managed File Transfer, implican canales de IBM MQ. Un canal de IBM MQ representa un extremo de un enlace de red. Los canales se clasifican como canales de mensajes o canales MQI.

"Configuración de un gestor de colas multiinstancia para que funcione con MFT" en la página 805
IBM WebSphere MQ 7.0.1 y posterior es acepta la creación de gestores de colas multiinstancia. Un gestor de colas multiinstancia se reinicia automáticamente en un servidor en espera. Managed File Transfer soporta la conexión a gestores de colas de agente multiinstancia, a un gestor de colas de coordinación multiinstancia y a un gestor de colas de mandatos multiinstancia.

Tareas relacionadas

"Configuración de gestores de colas de red de MFT" en la página 800
Si la red de Managed File Transfer incluye más de un gestor de colas de IBM MQ, estos gestores de colas de IBM MQ deben poder comunicarse de forma remota entre sí.

Referencia relacionada

fteSetupCoordination

Crear una estructura de transferencia de archivos de IBM MQ

Puede configurar una estructura de Managed File Transfer, basándose en un único agente conectado a un gestor de colas en la misma máquina.

Acerca de esta tarea

La configuración de MFT se almacena en una estructura de archivos, en la vía de acceso a datos de IBM MQ, en la máquina donde se ubicará el agente.

La configuración de ejemplo siguiente es para un gestor de colas de MFT en IBM MQ 8.0 denominado SAMPLECOORD (con la seguridad inhabilitada) y un único agente MFT denominado SAMPLEAGENT:

```
+--- config
             +--- SAMPLECOORD
                             +--- command.properties
                             +--- coordination.properties
                             +--- SAMPLECOORD.mqsc
                             +--- agents
                                          +--- SAMPLEAGENT
                                                       +--- agent.properties
                                                       +--- SAMPLEAGENT_create.mqsc
                                                       +--- SAMPLEAGENT_delete.mqsc

+---logs
             +--- SAMPLECOORD
                             +--- agents
                                          +--- SAMPLEAGENT
                                                       +--- logs
```

En este ejemplo, se supone que la seguridad del gestor de colas se ha inhabilitado. Los mandatos siguientes, que se ejecutan en **runmqsc**, inhabilitarán la seguridad después de que se reinicie el gestor de colas:

```
runmqsc queue manager
 alter qmgr CONNAUTH(NONE);
 alter qmgr CHLAUTH(DISABLED);
 end;
```

Para la configuración con la seguridad habilitada en MFT en IBM MQ 8.0 o posterior, **CONNAUTH** requiere que todos los mandatos de MFT que se conecten con un gestor de colas proporcionen credenciales de ID de usuario y contraseña. Puede aplicar los parámetros adicionales **-mquserid** y **-mqpassword** para cada mandato, o definir un archivo MQMFTCredentials.xml. El siguiente archivo de credenciales de

ejemplo define el ID de usuario `fteuser`, para el que se va a utilizar la contraseña `MyPassword` cuando se conecte al gestor de colas SAMPLECOORD:

```
<tns:mqmftCredentials xmlns:tns="http://wmqfte.ibm.com/MQMFTCredentials"
        xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://wmqfte.ibm.com/MQMFTCredentials MQMFTCredentials.xsd">
    <tns:qmgr mqPassword="MyPassword" MyUserId="fteuser" name="SAMPELCOORD"/>
</tns:mqmftCredentials>
```

Para obtener más información, consulte Autenticación de conexión de MFT y IBM MQ.

Notas:

- Para localizar su directorio de configuración de MFT, utilice el mandato **`fteDisplayVersion -v`**.
- z/OS Para los usuarios de z/OS , el archivo `MQMFTCredential.xml` se puede localizar como miembro en un conjunto de datos particionados con formato de registro variable (RECFM = V) o formato de registro no definido (RECFM = U).
- Para la configuración con la seguridad habilitada, añada el parámetro siguiente a los pasos siguientes para asociar las credenciales con el gestor de colas relevante: `-F` *`full_credential_file_path`*.
- La contraseña en texto simple en `MQMFTCredential.xml` se puede enmascarar utilizando el mandato siguiente:

  ```
  fteObfuscate -f full_file_path_to_MQMFTCredentials.xml
  ```

Procedimiento

1. Cree un gestor de colas de coordinación.

   Un gestor de colas de coordinación es un gestor de colas individual, que se utiliza para recibir toda la información de estado y registro de transferencias de sus agentes. Ejecute el siguiente mandato:

   ```
   fteSetupCoordination -coordinationQMgr coordination_qmgr_name
   ```

   Esto crea la configuración básica de nivel superior y crea un archivo de script IBM MQ para llamar a *`coordination_qmgr_name`*`.mqsc`.

   A continuación, la configuración deberá cargarse en el gestor de colas ejecutando el siguiente mandato de IBM MQ:

   ```
   runmqsc queue manager name < coordination_qmgr_name.mqsc
   ```

   Nota: Para la conexión de cliente TCP a un gestor de colas, puede utilizar:

   ```
   fteSetupCoordination -coordinationQMgr coordination_qmgr_name
   -coordinationQMgrHost coordination_qmgr_host -coordinationQMgrPort coordination_qmgr_port
   -coordinationQMgrChannel coordination_qmgr_channel
   ```

   Para el *`coordination_qmgr_name`*`.mqsc`creado, tendrá que ejecutar el mandato **`runmqsc`** en la misma máquina en la que se ejecuta el gestor de colas de coordinación.
2. Cree el gestor de colas de mandatos.

   Un gestor de colas de mandatos es un gestor de colas individual que se ha preconfigurado para que la infraestructura de IBM MQ pueda direccionar las solicitudes de MFT al agente correspondiente. Ejecute el siguiente mandato:

   ```
   fteSetupCommands -connectionQMgr Command QM Name -p Coordination QM Name
   ```

Con ello se crea un archivo `command.properties` en el directorio de coordinación. Tenga en cuenta que `-p` es opcional, y no es necesario si los mandatos se están configurando para la coordinación predeterminada.

Nota: Para la conexión de cliente TCP a un gestor de colas, puede utilizar:

```
fteSetupCommands -p coordination_qmgr_name -commandQMgr connection_qmgr_name
-commandQMgrHost connection_qmgr_host -commandQMgrPort connection_qmgr_port
-commandQMgrChannel connection_qmgr_channel
```

3. Cree el agente.

   Un agente es una aplicación que puede enviar y recibir archivos. Ejecute el siguiente mandato:

   ```
   fteCreateAgent -p coordination_qmgr_name -agentName agent_name -agentQMgr agent_qmgr_name
   ```

   Esto crea la configuración del agente bajo la coordinación y crea un archivo de script IBM MQ para llamar a *agent_name*`.mqsc` en el directorio de configuración del agente.

   Ejecute el siguiente mandato de IBM MQ para cargar el archivo de script de IBM MQ en el gestor de colas:

   ```
   runmqsc agent_qmgr_name < agent_name_create.mqsc file
   ```

   Nota: Para la conexión de cliente TCP a un gestor de colas, puede utilizar:

   ```
   fteCreateAgent -p coordination_qmgr_name -agentName agent_name -agentQMgr agent_qmgr_name
   -agentQMgrHost agent_qmgr_host -agentQMgrPort agent_qmgr_port -agentQMgrChannel
   agent_qmgr_channel
   ```

4. Inicie el agente.

   Ejecute el siguiente mandato:

   ```
   fteStartAgent -p coordination_qmgr_name agentName
   ```

   El agente se inicia en segundo plano y se devuelve el indicador de mandatos. Para comprobar que el agente se está ejecutando, emita el siguiente mandato:

   ```
   fteListAgents -p coordination_qmgr_name
   ```

   Esto muestra el estado de los agentes. Si el agente se está ejecutando correctamente, se notifica que está en el estado READY.

Resultados

Hay una infraestructura básica de MFT lista para su uso y ahora puede utilizar el mandato **fteCreateTransfer** para solicitar una transferencia. De forma alternativa, si IBM MQ Explorer está disponible, utilice los plugins de MFT para crear y supervisar las transferencias.

Pueden añadirse más agentes a la configuración repitiendo el paso 3 para crear un agente. Si se utiliza la conexión de cliente TCP, pueden estar en máquinas diferentes. Si están en máquinas diferentes, deben repetirse los mandatos **fteSetupCoordination** y **fteSetupCommands** para cada máquina, aunque no será necesario ejecutar los scripts mqsc.

Las configuraciones más complejas pueden tener varios gestores de colas para la coordinación y cada agente. En estos casos, los distintos gestores de colas deberán estar conectados entre ellos.

Conceptos relacionados

Qué hacer si el agente MFT no aparece en la lista del mandato **fteListAgents**

Referencia relacionada
fteSetupCoordination
fteSetupCommands: crear el archivo command.properties de MFT
fteCreateAgent
fteObfuscate: cifrar datos confidenciales
Formato del archivo de credenciales de MFT
El archivo MFT agent.properties

Configuración de un gestor de colas multiinstancia para que funcione con MFT

IBM WebSphere MQ 7.0.1 y posterior es acepta la creación de gestores de colas multiinstancia. Un gestor de colas multiinstancia se reinicia automáticamente en un servidor en espera. Managed File Transfer soporta la conexión a gestores de colas de agente multiinstancia, a un gestor de colas de coordinación multiinstancia y a un gestor de colas de mandatos multiinstancia.

Configuración de un gestor de colas multiinstancia

Importante: Si desea más información sobre cómo configurar un gestor de colas multiinstancia IBM MQ, consulte "Gestores de colas multiinstancia" en la página 530. Asegúrese de que ha leído esta información antes de intentar configurar un gestor de colas multiinstancia para que funcione con Managed File Transfer.

Utilización de un gestor de colas multiinstancia como gestor de colas de agente

Para que un agente se pueda conectar a la instancia activa y en espera del gestor de colas de varias instancias, añada la propiedad `agentQMgrStandby` al archivo `agent.properties` del agente. La propiedad `agentQMgrStandby` define el nombre de host y el número de puerto que se utilizan para las conexiones de cliente de la instancia del gestor de colas en espera. El valor de la propiedad debe proporcionarse en formato CONNAME de MQ , es decir, *host_name*(*port_number*).

La propiedad `agentQMgr` especifica el nombre del gestor de colas multiinstancia. La propiedad `agentQMgrHost` especifica el nombre de host de la instancia del gestor de colas activos y la propiedad `agentQMgrPort` especifica el número de puerto de la instancia del gestor de colas activo. El agente debe conectarse en modalidad de cliente a la instancia activa y en espera del gestor de colas multiinstancia.

Consulte El archivo MFT `agent.properties` para obtener más información.

Este ejemplo muestra el contenido del archivo `agent.properties` para AGENT1 que se conecta a un gestor de colas de varias instancias denominado QM_JUPITER. La instancia activa de QM_JUPITER se encuentra en el sistema host1 y utiliza el número de puerto 1414 para las conexiones de cliente. La instancia en espera de QM_JUPITER se encuentra en el sistema host2 y utiliza el número de puerto 1414 para las conexiones de cliente.

```
agentName=AGENT1
agentDesc=
agentQMgr=QM_JUPITER
agentQMgrPort=1414
agentQMgrHost=host1
agentQMgrChannel=SYSTEM.DEF.SVRCONN
agentQMgrStandby=host2(1414)
```

Utilización de un gestor de colas multiinstancia como gestor de colas de coordinación

Para habilitar las conexiones a la instancia activa y en espera del gestor de colas de coordinación multiinstancia, añada la propiedad `coordinationQMgrStandby` a todos los archivos `coordination.properties` de la topología de Managed File Transfer.

Consulte El archivo coordination.properties de MFT para obtener más información.

Este ejemplo muestra el contenido de un archivo `coordination.properties` que especifica los detalles de conexión a un gestor de colas de coordinación de varias instancias denominado QM_SATURN. La instancia activa de QM_SATURN se encuentra en el sistema coordination_host1 y utiliza el número de puerto 1420 para las conexiones de cliente. La instancia en espera de QM_SATURN se encuentra en coordination_host2 y utiliza el número de puerto 1420 para las conexiones de cliente.

```
coordinationQMgr=QM_SATURN
coordinationQMgrHost=coordination_host1
coordinationQMgrPort=1420
coordinationQMgrChannel=SYSTEM.DEF.SVRCONN
coordinationQMgrStandby=coordination_host2(1420)
```

El registrador autónomo de Managed File Transfer debe conectarse siempre al gestor de colas en modalidad de enlaces. Cuando utilice el registrador autónomo con un gestor de colas de coordinación multiinstancia, conecte el registrador autónomo, en modalidad de enlaces, a un gestor de colas diferente. Los pasos pertinentes se describen en "Configuraciones alternativas para un registrador autónomo de MFT" en la página 832. Debe definir los canales entre el gestor de colas del registrador autónomo y el gestor de colas de coordinación con el nombre de host y el número de puerto de ambas instancias del gestor de colas de coordinación multiinstancia. Si desea más información sobre cómo hacerlo, consulte "Gestores de colas multiinstancia" en la página 530.

El plug-in de Managed File Transfer para IBM MQ Explorer se conecta al gestor de colas de coordinación en modalidad de cliente. Si la instancia activa del gestor de colas de coordinación multiinstancia falla, la instancia en espera del gestor de colas de coordinación se activa y el plug-in se reconecta.

Los mandatos de Managed File Transfer **`fteList*`** y **`fteShowAgentDetails`** se conectan directamente con el gestor de colas de coordinación. Si la instancia activa de la coordinación multiinstancia no está disponible, estos mandatos intentarán conectarse a la instancia en espera del gestor de colas de coordinación.

Utilización de un gestor de colas multiinstancia como gestor de colas de mandatos

Para habilitar las conexiones a la instancia activa y en espera del gestor de colas de coordinación multiinstancia, añada la propiedad `connectionQMgrStandby` a todos los archivos `command.properties` de la topología de Managed File Transfer.

Consulte El archivo command.properties de MFT para obtener más información.

Este ejemplo muestra el contenido de un archivo `command.properties` que especifica los detalles de conexión a un gestor de colas de mandatos de varias instancias denominado QM_MARS. La instancia activa de QM_MARS se encuentra en el sistema command_host1 y utiliza el número de puerto 1424 para las conexiones de cliente. La instancia en espera de QM_MARS se encuentra en el sistema command_host2 y utiliza el número de puerto 1424 para las conexiones de cliente.

```
connectionQMgr=QM_SATURN
connectionQMgrHost=command_host1
connectionQMgrPort=1424
connectionQMgrChannel=SYSTEM.DEF.SVRCONN
connectionQMgrStandby=command_host2(1424)
```

Conceptos relacionados

"conexión a IBM MQ" en la página 798
Todas las comunicaciones de red con los gestores de colas de IBM MQ, incluyendo la comunicación relacionada con Managed File Transfer, implican canales de IBM MQ. Un canal de IBM MQ representa un extremo de un enlace de red. Los canales se clasifican como canales de mensajes o canales MQI.

Tareas relacionadas

"Configuración de gestores de colas de red de MFT" en la página 800
Si la red de Managed File Transfer incluye más de un gestor de colas de IBM MQ, estos gestores de colas de IBM MQ deben poder comunicarse de forma remota entre sí.

"Configuración del gestor de colas de coordinación para MFT" en la página 801

Después de ejecutar el mandato **fteSetupCoordination** , ejecute el script *coordination_qmgr_name*.mqsc en el directorio *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name* para realizar la configuración necesaria para el gestor de colas de coordinación. Sin embargo, si desea realizar esta configuración manualmente, realice los pasos siguientes en el gestor de colas de coordinación.

Conservación de mensajes de registro de MFT

Managed File Transfer envía información de progreso y de registro de la transferencia de archivos al gestor de colas de coordinación. El gestor de colas de coordinación publica esta información en las suscripciones coincidentes al tema SYSTEM.FTE. Si no existe ninguna suscripción, esta información no se conserva.

Formas de garantizar que se conserva la información

Si la información de progreso o registro de la transferencia es importante para su empresa, debe ejecutar uno de los pasos siguientes para garantizar que se conserva esta información:

- Utilice el registrador de base de datos Managed File Transfer para copiar los mensajes publicados en SYSTEM.FTE/Log en una base de datos Oracle o Db2 .
- Defina una suscripción a SYSTEM.FTE , que almacena publicaciones en una cola IBM MQ . Defina esta suscripción antes de transferir transferencias de archivos para garantizar que todos los mensajes de progreso y de registro se conservan en la cola.
- Escriba una aplicación que utilice la interfaz de colas de mensajes (MQI) o IBM MQ JMS para crear una suscripción duradera y procesar las publicaciones que se suministran a la suscripción. Esta aplicación debe estar en funcionamiento antes de transferir archivos para garantizar que la aplicación recibe todos los mensajes de progreso y de registro.

Cada uno de estos métodos se describe con mayor detalle en los apartados que vienen a continuación.

No utilice el plug-in de IBM MQ Explorer para conservar información de registro.

Utilizar el registrador de base de datos de Managed File Transfer para conservar mensajes de registro

El registrador de base de datos es un componente opcional de Managed File Transfer que se puede utilizar para copiar información de registro en una base de datos para fines de análisis y auditoría. El registrador de base de datos es una aplicación autónoma Java que se instala en un sistema que aloja el gestor de colas de coordinación y la base de datos. Para obtener más información sobre el registrador de base de datos, consulte “Configuración de un registrador de MFT” en la página 819.

Conservación de mensajes de progreso y registro utilizando el plug-in IBM MQ Explorer

Cuando se inicia por primera vez una instancia del plug-in de IBM MQ Explorer, la instancia crea una suscripción duradera en el gestor de colas de coordinación. Esta suscripción duradera se utiliza para recopilar la información que se visualiza en las vistas **Registro de transferencias** y **Proceso de la transferencia actual**.

El nombre de la suscripción duradera se añade como prefijo para mostrar que la suscripción ha sido creada por el plug-in de IBM MQ Explorer MFT , el nombre de host y el nombre del usuario. Por ejemplo, `MQExplorer_MFT_Plugin_HOST_TJWatson`.

Este prefijo se añade en caso de que un administrador desee suprimir una descripción duradera que ya no esté activa mediante una instancia del plug-in de IBM MQ Explorer.

La utilización de una suscripción duradera en el gestor de colas de coordinación puede hacer que se acumulen mensajes en las colas SYSTEM.MANAGED.DURABLE. Si tiene una red de Managed File Transfer de gran volumen, utiliza el plug-in de IBM MQ Explorer con poca frecuencia, o ambas cosas, estos datos de mensajes pueden llenar el sistema de archivos local.

Para evitar que esto suceda, especifique que el plug-in de IBM MQ Explorer utilice una suscripción no duradera al gestor de colas de coordinación. Realice los pasos siguientes en IBM MQ Explorer:

1. Seleccione **Ventana** > **Preferencias** > **MQ Explorer** > **Managed File Transfer**
2. En la lista **Transfer Log subscription type** (Tipo de suscriptor de Registro de transferencias), elija NON_DURABLE.

Almacenamiento de publicaciones en una cola de IBM MQ

Para almacenar mensajes de registro o progreso en una cola de IBM MQ, configure una suscripción en el gestor de colas de coordinación que reenvía mensajes a esta cola. Por ejemplo, para reenviar todos los mensajes de registro a una cola denominada LOG.QUEUE, someta el siguiente mandato MQSC:

```
define sub(MY.SUB) TOPICSTR('Log/#') TOPICOBJ(SYSTEM.FTE) DEST(LOG.QUEUE)WSCHEMA(TOPIC)
```

Cuando se hayan reenviado los mensajes de registro a una cola de IBM MQ, permanecerán en ella hasta que los haya procesado una aplicación de IBM MQ que la utiliza.

Escritura de aplicaciones que gestionan una suscripción duradera al tema SYSTEM.FTE

Puede escribir aplicaciones que gestionen sus propias suscripciones duraderas en SYSTEM.FTE utilizando una de las interfaces de programación de aplicaciones soportadas por IBM MQ. Estas aplicaciones pueden recibir mensajes de cola o de registro de IBM MQ y actuar debidamente para satisfacer las necesidades de la empresa.

Para obtener más información sobre las interfaces de programación de aplicaciones disponibles, consulte Desarrollo de aplicaciones.

Configuración de gestores de colas de agente de MFT

Después de la instalación, ejecute el script *agent_name*_create.mqsc en el directorio *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name*/agents/*agent_name* para realizar la configuración necesaria para el gestor de colas del agente. Sin embargo, si desea realizar esta configuración manualmente, complete estos pasos en el gestor de colas de agente.

Procedimiento

1. Cree las colas de operaciones del agente.

   Estas colas se denominan:

   - SYSTEM.FTE.COMMAND.*nombre_agente*
   - SYSTEM.FTE.DATA.*nombre_agente*
   - SYSTEM.FTE.EVENT.*nombre_agente*
   - SYSTEM.FTE.REPLY.*nombre_agente*
   - SYSTEM.FTE.STATE.*nombre_agente*

   Para obtener información sobre los parámetros de cola y cómo se utilizan las colas, consulte Valores de cola de agente deMFT.
2. Cree las colas de autorización del agente.

   Estas colas se denominan:

   - SYSTEM.FTE.AUTHADM1.*nombre_agente*
   - SYSTEM.FTE.AUTHAGT1.*nombre_agente*
   - SYSTEM.FTE.AUTHMON1.*nombre_agente*
   - SYSTEM.FTE.AUTHOPS1.*nombre_agente*

- SYSTEM.FTE.AUTHSCH1.*nombre_agente*
- SYSTEM.FTE.AUTHTRN1.*nombre_agente*

Para obtener información sobre los parámetros de cola y cómo se utilizan las colas, consulte Valores de cola de agente deMFT.

Qué hacer a continuación

Para obtener información sobre la creación y configuración de un agente de puente de protocolo, consulte fteCreateBridgeAgent (crear y configurar un agente de puente de protocolo MFT) y Configuración de un puente de protocolo para un servidor FTPS.

Conceptos relacionados

"conexión a IBM MQ" en la página 798
Todas las comunicaciones de red con los gestores de colas de IBM MQ, incluyendo la comunicación relacionada con Managed File Transfer, implican canales de IBM MQ. Un canal de IBM MQ representa un extremo de un enlace de red. Los canales se clasifican como canales de mensajes o canales MQI.

"Configuración de un gestor de colas multiinstancia para que funcione con MFT" en la página 805
IBM WebSphere MQ 7.0.1 y posterior es acepta la creación de gestores de colas multiinstancia. Un gestor de colas multiinstancia se reinicia automáticamente en un servidor en espera. Managed File Transfer soporta la conexión a gestores de colas de agente multiinstancia, a un gestor de colas de coordinación multiinstancia y a un gestor de colas de mandatos multiinstancia.

Tareas relacionadas

"Configuración de gestores de colas de red de MFT" en la página 800
Si la red de Managed File Transfer incluye más de un gestor de colas de IBM MQ, estos gestores de colas de IBM MQ deben poder comunicarse de forma remota entre sí.

"Configuración del gestor de colas de coordinación para MFT" en la página 801
Después de ejecutar el mandato **fteSetupCoordination** , ejecute el script *coordination_qmgr_name*.mqsc en el directorio *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name* para realizar la configuración necesaria para el gestor de colas de coordinación. Sin embargo, si desea realizar esta configuración manualmente, realice los pasos siguientes en el gestor de colas de coordinación.

Referencia relacionada

Valores de cola del agente MFT
fteSetupCoordination

Configuración de un agente MFT para varios canales en un clúster

Si desea utilizar el soporte multicanal de IBM MQ en una configuración en clúster, primero establezca la propiedad **agentMultipleChannelsEnabled** en `true` y, a continuación, complete los pasos de este tema.

Acerca de esta tarea

En un clúster, el soporte de varios canales está habilitado sólo por las definiciones de IBM MQ en el gestor de colas del agente de destino.

Debe realizar los pasos de este tema, además de los pasos de configuración estándar de IBM MQ necesarios para un agente de Managed File Transfer, que se listan en "Configuración de MFT cuando se utiliza por primera vez" en la página 798.

Los ejemplos de configuración siguientes utilizan mandatos **runmqsc**.

Procedimiento

1. Defina un canal de clúster receptor para cada canal que desee utilizar. Por ejemplo, si está utilizando dos canales:

```
DEFINE CHANNEL(TO.DESTQMGRNAME_1) CHLTYPE(CLUSRCVR) CLUSTER(MFTCLUSTER)
DEFINE CHANNEL(TO.DESTQMGRNAME_2) CHLTYPE(CLUSRCVR) CLUSTER(MFTCLUSTER)
```

donde:

- *DESTQMGRNAME* es el nombre del gestor de colas del agente de destino.
- *MFTCLUSTER* es el nombre del clúster de IBM MQ.

Se recomienda utilizar el convenio de denominación *MFTCLUSTER*.*DESTMGRNAME*_n para canales, pero este convenio no es obligatorio.

2. Defina un alias de gestor de colas correspondiente a cada canal. Por ejemplo:

```
DEFINE QREMOTE(SYSTEM.FTE.DESTQMGRNAME_1) RQMNAME(DESTQMGRNAME) CLUSTER(MFTCLUSTER)
DEFINE QREMOTE(SYSTEM.FTE.DESTQMGRNAME_2) RQMNAME(DESTQMGRNAME) CLUSTER(MFTCLUSTER)
```

Debe utilizar SYSTEM.FTE.*DESTQMGRNAME_n* convenio de denominación para alias de gestor de colas porque el agente emisor busca alias de gestor de colas de este formato. Los números que utilice para *n* deben comenzar en 1 y ser consecutivos. Debe hacer las definiciones para todo el clúster para que estén disponibles en el gestor de colas del agente de origen.

Para que tanto el agente de origen como el de destino determinen correctamente el número de alias de gestor de colas, **no** defina una XMITQ predeterminada para el gestor de colas.

Tareas relacionadas

“Configuración de un agente MFT para varios canales: no en clúster” en la página 810
Si desea utilizar el soporte de varios canales de IBM MQ en una configuración no en clúster, primero establezca la propiedad agentMultipleChannelsEnabled en `true` y, a continuación, complete los pasos de este tema.

Referencia relacionada

El archivo MFT `agent.properties`

Configuración de un agente MFT para varios canales: no en clúster

Si desea utilizar el soporte de varios canales de IBM MQ en una configuración no en clúster, primero establezca la propiedad agentMultipleChannelsEnabled en `true` y, a continuación, complete los pasos de este tema.

Acerca de esta tarea

En una configuración no en clúster, las definiciones de IBM MQ habilitan el soporte de varios canales en el gestor de colas tanto del agente de origen como del agente de destino.

Debe realizar los pasos de este tema, además de los pasos de configuración estándar de IBM MQ necesarios para un agente de Managed File Transfer, que se listan en “Configuración de MFT cuando se utiliza por primera vez” en la página 798.

Los pasos siguientes presuponen que se están utilizando canales emisores-receptores para la comunicación entre los gestores de colas de origen y de destino.

Los ejemplos de configuración siguientes utilizan mandatos **runmqsc**.

Procedimiento

1. En el gestor de colas del agente de destino, defina un canal receptor para cada canal que desee utilizar. Por ejemplo, si está utilizando dos canales:

```
DEFINE CHANNEL(TO.DESTQMGRNAME_1) CHLTYPE(RCVR) TRPTYPE(TCP)
DEFINE CHANNEL(TO.DESTQMGRNAME_2) CHLTYPE(RCVR) TRPTYPE(TCP)
```

donde: DESTQMGRNAME es el nombre del gestor de colas del agente de destino.

Se recomienda utilizar el convenio de denominación TO.DESTMGRNAME_n para los canales, pero este convenio no es obligatorio. Los nombres de los canales receptores deben coincidir con los canales emisores correspondientes en el gestor de colas del agente de origen.

2. En el gestor de colas del agente de origen, defina una cola de transmisión para cada canal que desee utilizar. Por ejemplo, si está utilizando dos canales:

```
DEFINE QLOCAL(DESTQMGRNAME_1) USAGE(XMITQ)
DEFINE QLOCAL(DESTQMGRNAME_2) USAGE(XMITQ)
```

   Se recomienda utilizar el convenio de denominación DESTMGRNAME_n para las colas de transmisión, pero este convenio no es obligatorio. Las colas de transmisión que defina son referenciadas desde las definiciones de canal emisor y las definiciones de alias de gestor de colas en los pasos siguientes.

3. En el gestor de colas del agente de origen, defina un canal emisor para cada canal que desee utilizar. Por ejemplo, si está utilizando dos canales:

```
DEFINE CHANNEL(TO.DESTQMGRNAME_1) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(DESTHOST:port)
XMITQ(DESTQMGRNAME_1)
DEFINE CHANNEL(TO.DESTQMGRNAME_2) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(DESTHOST:port)
XMITQ(DESTQMGRNAME_2)
```

   Se recomienda utilizar el convenio de denominación TO.DESTMGRNAME_n para los canales, pero este convenio no es obligatorio. Los nombres de los canales emisores deben coincidir con los canales receptores correspondientes en el gestor de colas del agente de destino.

4. En el gestor de colas del agente de origen, defina un alias de gestor de colas correspondiente a cada canal. Por ejemplo:

```
DEFINE QREMOTE(SYSTEM.FTE.DESTQMGRNAME_1) RQMNAME(DESTQMGRNAME) XMITQ(DESTQMGRNAME_1)
DEFINE QREMOTE(SYSTEM.FTE.DESTQMGRNAME_2) RQMNAME(DESTQMGRNAME) XMITQ(DESTQMGRNAME_2)
```

   Debe utilizar el convenio de denominación SYSTEM.FTE.DESTQMGRNAME_n para los alias de gestor de colas, ya que el agente emisor busca alias de gestor de colas de este formato. Los números que utilice para *n* deben comenzar en 1 y ser consecutivos.

   Para que el agente determine correctamente el número de alias de gestor de colas, **no** defina una XMITQ predeterminada para el gestor de colas.

Tareas relacionadas

“Configuración de un agente MFT para varios canales en un clúster” en la página 809
Si desea utilizar el soporte multicanal de IBM MQ en una configuración en clúster, primero establezca la propiedad **agentMultipleChannelsEnabled** en `true` y, a continuación, complete los pasos de este tema.

Referencia relacionada

El archivo MFT `agent.properties`

Windows Configuración de agentes de MFT con MSCS

La configuración del Managed File Transfer agente (MFT) Microsoft Cluster Service (MSCS) está soportada, si la plataforma es una soportada por MFT y ejecuta una de las versiones de Windows.

Acerca de esta tarea

Esta tarea describe dos escenarios que puede seguir para conseguir la migración tras error de un agente MFT:

- Escenario 1: configuración del agente como un recurso MSCS.
- Escenario 2: configuración del gestor de colas de agente y el agente como recursos MSCS.

Procedimiento

Escenario 1: configuración del agente como un recurso MSCS

- Para configurar el agente como un recurso MSCS, complete los pasos siguientes:
  a) Instale Managed File Transfer localmente en cada máquina en la agrupación.

     Consulte Instalación de Managed File Transfer.

  b) Cree el agente en la máquina primaria del clúster.

     El agente se debe configurar para conectarse al gestor de colas de agente utilizando el transporte CLIENT. Asegúrese de que crea todos los objetos en el gestor de colas para este agente. Para obtener información sobre cómo hacerlo, consulte Configuración del agente.

  c) Modifique el agente para que se ejecute como un servicio de Windows, y configúrelo para que no se inicie automáticamente cuando se reinicia Windows estableciendo el campo **Tipo de inicio** para el servicio del agente en la herramienta Servicios Windows en `Manual`.

     Para obtener más información. consulte Inicio de un agente MFT como un servicio de Windows.

  d) Repita el paso “2” en la página 812 y el paso “3” en la página 812 del Escenario 1 en la máquina secundaria.

     Esto garantiza que la estructura de archivos para los registros, propiedades, etc. existe en la otra máquina del clúster. Tenga en cuenta que no es necesario crear los objetos del gestor de colas como en el paso “2” en la página 812.

  e) En la máquina primaria, añada el agente como un 'Servicio genérico' bajo el control de MSCS.

     Para ello:

     a. Pulse con el botón derecho del ratón en el clúster y seleccione **Rol -> Añadir recurso -> 'Servicio genérico'**.

     b. En la lista de servicios de Windows, seleccione el servicio de agente y complete el asistente de configuración pulsando **Siguiente**.

     Ahora el servicio de agente se ha añadido como un recurso MSCS. Si se produce una migración tras error, el servicio de agente se iniciará en la otra máquina.

Escenario 2: configuración del gestor de colas de agente y el agente como recursos MSCS

- Para configurar el gestor de colas de agente y el agente como recursos MSCS, complete los pasos siguientes:
  a) Configure el gestor de colas de agente para ejecutarse como un recurso MSCS.

     Si desea más información sobre cómo hacerlo, consulte “Poner un gestor de colas bajo control de MSCS” en la página 518.

  b) Cree el agente en la máquina primaria del clúster.

     El agente se debe configurar para conectarse al gestor de colas de agente utilizando el transporte BINDINGS. Asegúrese de que crea todos los objetos en el gestor de colas para este agente. Para obtener información sobre cómo hacerlo, consulte Configuración del agente.

  c) Modifique el agente para que se ejecute como un servicio de Windows, y configúrelo para que no se inicie automáticamente cuando se reinicia Windows estableciendo el campo **Tipo de inicio** para el servicio del agente en la herramienta Servicios Windows en `Manual`.

     Para obtener más información. consulte Inicio de un agente MFT como un servicio de Windows.

  d) Asegúrese de que el gestor de colas de agente (que está bajo el control de MSCS) se está ejecutando en la máquina secundaria.

     El agente que se crea en esta máquina se conectará al gestor de colas utilizando el transporte BINDINGS, por lo tanto, tiene que estar disponible cuando se crea el agente.

  e) Repita el paso “2” en la página 812 y el paso “3” en la página 812 del escenario 2 en la máquina secundaria.

     Esto garantiza que la estructura de archivos para los registros, propiedades, etc. existe en la otra máquina del clúster. Tenga en cuenta que no es necesario crear los objetos del gestor de colas como en el paso “2” en la página 812.

f) Añada el agente como un 'Servicio genérico' bajo el control de MSCS.

Para ello:

a. Pulse con el botón derecho del ratón en el clúster y seleccione **Rol -> Añadir recurso -> 'Servicio genérico'**.

b. En la lista de servicios de Windows, seleccione el servicio de agente y complete el asistente de configuración pulsando **Siguiente**.

g) Modifique las propiedades de recursos del servicio de agente para añadir el recurso del gestor de colas en la lista de dependencias.

Esto garantiza que el recurso del gestor de colas se inicia antes de que se inicie el agente.

h) Ponga el recurso del gestor de colas fuera de línea y, después, ponga el recurso de agente en línea. Verifique si se han iniciado ambos, el recurso del gestor de colas y el agente.

Si se produce una migración tras error, el servicio de agente y el gestor de colas de agente se iniciarán en la máquina secundaria.

Agentes de alta disponibilidad en Managed File Transfer

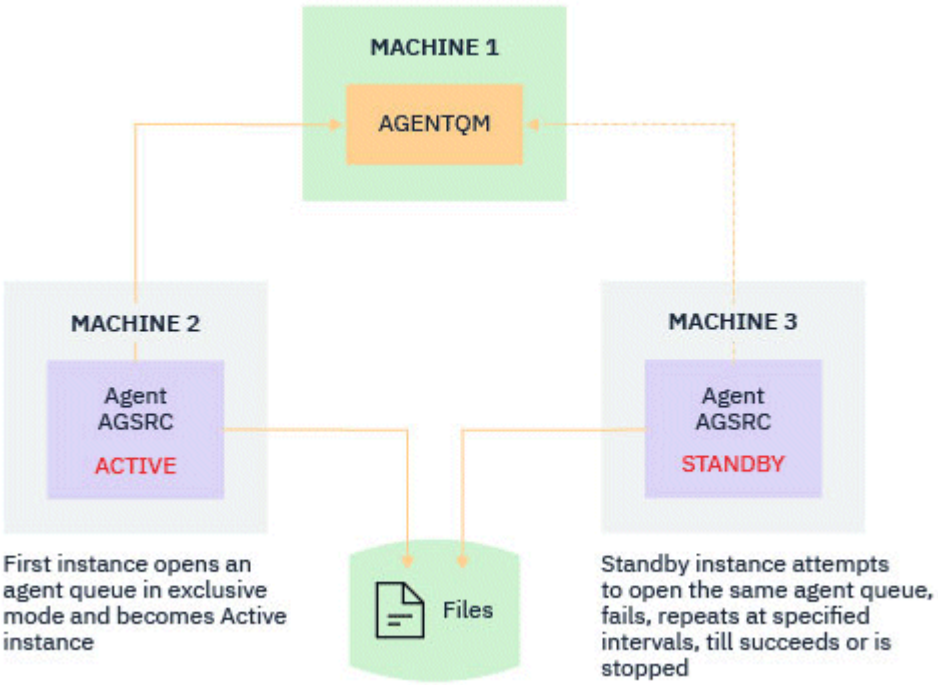
Puede configurar agentes estándar o de puente en MFT para que se ejecutan en una configuración de alta disponibilidad (HA). Un par de instancias de agente con configuraciones idénticas están implicadas en la configuración de HA, donde una instancia se está ejecutando en una máquina, mientras que la otra instancia se está ejecutando en una máquina distinta. Ambas instancias se han configurado para conectarse al mismo gestor de colas de agente,

Visión general

Solo una de las dos instancias, llamada *instancia activa*, procesa las transferencias de archivos, mientras que la otra instancia, llamada *instancia en espera*, está en un estado parcialmente inicializado y no puede procesar ninguna transferencia de archivos.

Cuando una instancia activa falla o pierde la conectividad con el gestor de colas, la instancia en espera completa su inicialización, se convierte en la instancia activa e inicia el proceso de las transferencias de archivos. Las transferencias en curso que lo estaban cuando falló la instancia activa se reanudan a partir del último punto de comprobación conocido.

La ilustración siguiente muestra una configuración común de los agentes activos y en



espera:

Notas:

1. Una instancia de un agente se está ejecutando en dos máquinas diferentes, con una de las instancias como una *instancia activa* y la otra como la *instancia en espera*.
2. Una instancia de un agente se está ejecutando en una máquina distinta, con una de las instancias como instancia activa y la otra como la instancia en espera.
3. El mismo conjunto de colas de agente se comparte entre dos instancias del agente.
4. Ambas instancias del agente necesitan acceder al mismo sistema de archivos compartidos para realizar transferencias gestionadas.

El mecanismo de la instancia de agente activa-en espera funciona realizando un bloqueo en un recurso compartido. La instancia del agente que toma un bloqueo en el recurso compartido se convierte en la instancia activa, mientras que la otra instancia (que no puede tomar un bloqueo) se convierte en una instancia en espera.

El recurso compartido aquí es una cola nueva, `SYSTEM.FTE.HA.<agent name>`. Esta cola se crea automáticamente cuando se configura un agente de IBM MQ 9.1.4 o posterior.

Cómo funciona el proceso

Para crear un agente HA, cree un agente con parámetros de configuración idénticos en dos máquinas ejecutando el mandato **`fteCreateAgent`** o **`fteCreateBridgeAgent`** utilizando el parámetro **`-x`** adicional junto con la propiedad de agente **`highlyAvailable`** en el archivo `agent.properties` establecida en `true`.

Notas:

- Ambas configuraciones deben apuntar al mismo gestor de colas de agente.
- Las colas de agente necesarias solo se deben crear una vez en el gestor de colas de agente.

Consulte el mandato **`fteCreateAgent`** para obtener más información sobre el parámetro **`-x`** y el archivo `agent.properties`, para obtener más información sobre la propiedad del agente **`highlyAvailable`**.

Nota: Al ejecutar el mandato **`fteCreateAgent`** o **`fteCreateBridgeAgent`** se crea un archivo MQSC que contiene los scripts necesarios para crear objetos IBM MQ en el gestor de colas del agente y la cola

SYSTEM.FTE.HA.*agent name* . Este archivo MQSC se crea tanto si especifica el parámetro **-x** como si no.

Al crear una configuración de agente de alta disponibilidad, el mandato **fteCreateAgent** o **fteCreateBridgeAgent** comprueba la existencia de una instancia del mismo agente presente en otro lugar suscribiéndose al tema SYSTEM.FTE/Agents/*agent name* . Si se encuentra una instancia del mismo agente, cualquier de estos mandatos crea la configuración necesaria en el sistema de archivos, pero no vuelve a publicar la creación del agente.

Cuando un agente se inicia en modalidad HA:

1. El agente intenta abrir la cola SYSTEM.FTE.HA.*agent name* en una modalidad GET exclusiva.
2. Si el agente abre la cola SYSTEM.FTE.HA.*agent name* correctamente, se convierte en la *instancia activa* de un agente y continúa un proceso de inicio adicional.
3. Si el intento de abrir la cola SYSTEM.FTE.HA.*agent name* en una modalidad GET exclusiva falla con el código de razón MQRC_OBJECT_IN_USE, significa que ya hay una instancia activa del agente que se ejecuta en otro lugar. Por lo tanto, esta instancia se convierte en la *instancia en espera* del agente.

   La instancia en espera intenta abrir la cola SYSTEM.FTE.HA.*agent name* a intervalos especificados. Se proporciona una propiedad de agente adicional **standbyPollInterval** para esta finalidad en el archivo agent.properties.

   Con el valor predeterminado, la instancia en espera intenta abrir la cola SYSTEM.FTE.HA.*agent name* cada cinco segundos. Esto se repite hasta que la instancia logra abrir la cola SYSTEM.FTE.HA.*agent name* o se detiene utilizando el mandato **fteStopAgent** .

   La propiedad **standbyPollInterval** también la utilizan todas las instancias para determinar cuánto tiempo espera una instancia entre intentos de reconexión si se desconecta de su gestor de colas de agente.

Varias instancias en espera

Todas las instancias en espera intentan tomar la cola SYSTEM.FTE.HA.*agent name* en una modalidad GET exclusiva, y la instancia que tiene éxito, después de que la instancia activa falle, se convierte en la instancia activa.

La instancia activa mantiene la información de todas las instancias en espera conocidas y publica la información como parte de la publicación del estado del agente. La salida del mandato **fteShowAgentDetails**, la respuesta GET REST API del agente y el conector IBM MQ Explorer MFT muestran información sobre todas las instancias en espera.

Consulte los resultados de ejemplo del mandato de **fteShowAgentDetails** y la respuesta de la GET REST API para obtener más información.

Consulte Mensajes de estado de agente MFT para ver ejemplos de información de estado de agente en formato XML.

Requisito de versión

Los agentes activo y en espera deben ser IBM MQ 9.1.4 o superiores.

Atención:

- No puede configurar ni iniciar versiones de IBM MQ anteriores a IBM MQ 9.1.4 en modalidades de alta disponibilidad.
- Ambas instancias, activa y en espera, deben ejecutar la misma versión del código.

La versión de las instancias activa y en espera se valida para asegurarse de que las dos instancias sean de la misma versión. Se utiliza una cola dinámica temporal para la comunicación entre las instancias. Dos propiedades de agente, **dynamicQueuePrefix** y **modelQueueName**, definidas en el archivo agent.properties, generan el nombre de la cola dinámica temporal.

Información necesaria para agentes de alta disponibilidad en Managed File Transfer

Hay varios tipos de información que necesita conocer acerca de los agentes de MFT estándar o de puente que se ejecutan en una configuración de alta disponibilidad. Esta información incluye distintos métodos mediante los que se inicia el agente, cómo identificar la instancia del agente en el archivo de registro y la información de estado del agente.

Inicio de un agente

Una instancia de un agente se está ejecutando en una modalidad no HA en algún lugar

Si se intenta iniciar otra instancia del agente que no está configurada como agente de alta disponibilidad (HA), primero se realiza una comprobación para ver si se puede adquirir un bloqueo en la cola de SYSTEM.FTE.HA.*agent name*.

Como la otra instancia se ha iniciado en modalidad de no HA, el bloqueo de la cola SYSTEM.FTE.HA.*agent name* será adquirido por esta instancia. El agente continúa la inicialización, pero falla, en un punto posterior, porque la cola de mandatos ha sido abierta exclusivamente por otra instancia.

En este caso, los mensajes que se muestran en el ejemplo siguiente se registran en el archivo output0.log del agente y el agente continúa su intento de abrir la cola de mandatos cada 30 segundos:

```
BFGMQ1045I: la cola de sistema del agente 'SYSTEM.FTE.COMMAND.SRC' está configurada como NOSHARE
o
DEFSOPT (SHARED).

BFGAG0035W: El agente ha recibido el código de razón MQI 2042 al intentar abrir la cola
'SYSTEM.FTE.COMMAND.SRC' en el gestor de colas 'MFTHAQM' con el nombre de conexión
'localhost(1414)'
y el canal 'MFT_HA_CHN'.  El agente volverá a intentar la operación cada 30 segundos.
```

Una instancia de un agente se está ejecutando en una modalidad HA en otra parte

Si se intenta iniciar otra instancia del agente que no está configurada como agente de alta disponibilidad (HA), primero se realiza una comprobación para ver si se puede adquirir un bloqueo en la cola de SYSTEM.FTE.HA.*agent name*.

Puesto que la otra instancia se ha estado ejecutando como una instancia activa, el intento de adquirir un bloqueo falla. La instancia no se puede iniciar y el siguiente mensaje de error se registra en el archivo output0.log del agente:

```
BFGAG0194E: Una instancia de este agente ya se está ejecutando en otra parte.
Por lo tanto, la instancia no puede continuar y terminará.
```

Windows Inicio del agente como un servicio de Windows

En Windows, puede iniciar un agente como un servicio de Windows.

Durante el inicio, Windows inicia el agente de MFT en modalidad normal o HA. Si el agente se ha configurado para iniciarse en la modalidad HA, el servicio se ejecuta como una instancia activa o en espera, en función la instancia que adquiera primero el bloqueo.

Identificación del tipo de instancia de un agente en el archivo de registro

Los mensajes de información se escriben en el archivo output0.log del agente para indicar el tipo de instancia. Cuando una instancia de agente se inicia como una instancia activa, se escribe el mensaje siguiente:

```
BFGAG0193I: el agente se ha inicializado correctamente como una instancia activa.
```

Cuando una instancia de agente se inicia como una instancia en espera, se escribe el mensaje siguiente:

```
BFGAG0193I: el agente se ha inicializado correctamente como una instancia en espera.
```

Actualizaciones de estado de agente

Puesto que hay dos instancias del mismo agente en ejecución, debe tener la información sobre ambas instancias en la publicación del estado de agente.

Tenga en cuenta que la instancia activa es la que está publicando el estado de ambas instancias.

Instancia en espera

Al publicar el estado del agente, la instancia activa comprueba la antigüedad de la publicación de la instancia en espera.

Hay dos propiedades adicionales en el archivo `agent.properties` para esta finalidad:

- **`standbyStatusExpiry`** es el tiempo de caducidad del mensaje de estado en espera que se debe colocar en la cola de mandatos del agente. El mensaje caduca si la instancia activa de un agente no procesa este mensaje en dicho periodo.

  De forma predeterminada, el valor de **`standbyStatusExipry`** es 30 segundos. El mensaje también es una prioridad baja, 9, mensaje para permitir el proceso de prioridad de las solicitudes de transferencia sobre los mensajes de estado en espera.
- **`standbyStatusPublishInterval`** establece la frecuencia a la que la instancia en espera publica su estado.

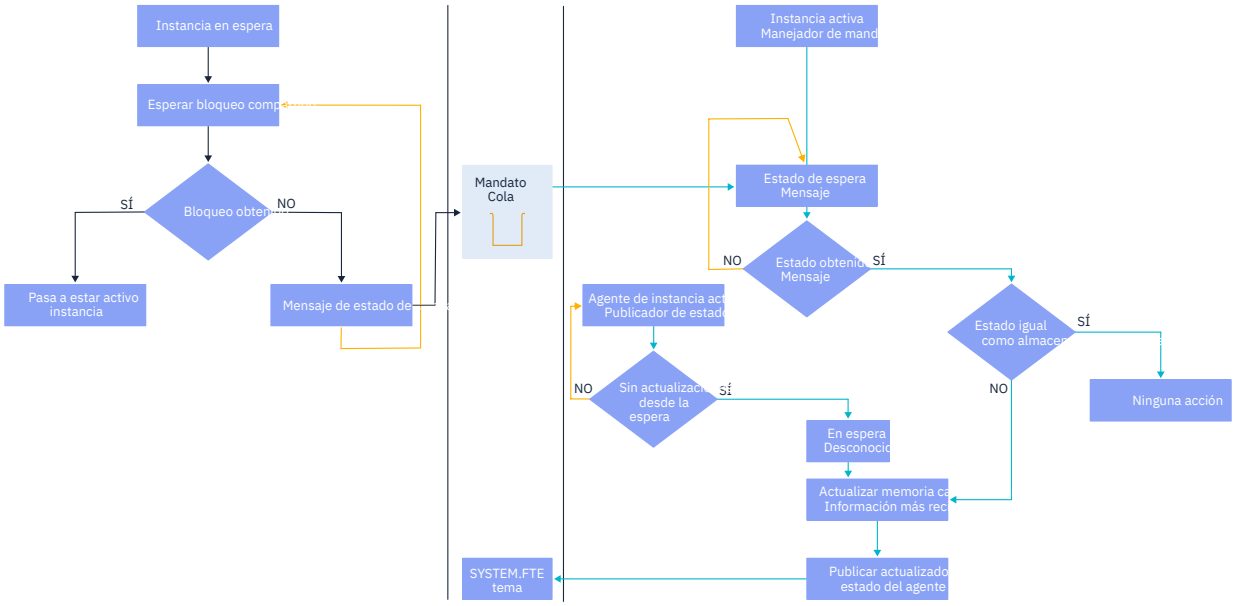
Instancia activa

La instancia activa realiza lo siguiente para procesar las actualizaciones de estado desde la instancia en espera:

1. Obtiene el mensaje de la cola SYSTEM.FTE.COMMAND.<agent name> y delega el proceso del mensaje en una hebra de trabajo.
2. La hebra de trabajador recupera el contenido del cuerpo del mensaje, actualiza el objeto de estado del agente con información de la instancia en espera y notifica al publicador del estado del agente que publique el estado.
3. El publicador de estado del agente publica el estado.

   Tenga en cuenta que las optimizaciones se realizan aquí para almacenar en memoria caché la información de estado en espera. Cuando se realiza una solicitud, el publicador de estado de agente comprueba el estado nuevo con el estado almacenado en la memoria caché y solo lo publica si hay una diferencia.

El diagrama siguiente describe el flujo de las instancias activas y en espera después de publicar el estado de un agente:



Descartar instancias, migración tras error y mantenimiento en agentes altamente disponibles

Las instancias de Managed File Transfer altamente disponibles se pueden descargar, pueden fallar de varias formas y pueden necesitar mantenimiento.

Descartar el estado de instancia en espera

Puede haber situaciones en las que la instancia activa está ocupada con transferencias y no puede procesar mensajes de estado de instancia en espera, o la instancia en espera ha fallado o no publica mensajes de estado por algún motivo.

En estos casos, el agente activo que estaba al tanto de la presencia de la instancia en espera espera el valor especificado por la propiedad **standbyStatusDiscardTime** en el archivo `agent.properties` antes de eliminar la instancia en espera de la lista. El valor predeterminado para esta propiedad es 600 segundos, que es dos veces el de la propiedad **standbyStatusPublishInterval**.

Migración tras error normal de a una instancia

Debe utilizar el mandato de **fteStopAgent** con la opción **-i** para realizar una migración tras error normal.

Esto garantiza que la instancia activa se detiene inmediatamente. Si detiene un agente sin la opción **-i**, el agente se sigue ejecutando hasta que todas las transferencias en curso hayan sido completadas por la instancia activa, por lo tanto, la migración tras error podría tardar mucho tiempo.

Las transferencias en curso se reanudan desde el último punto de comprobación conocido.

Migración tras error de una instancia en otras situaciones

Si una instancia activa finaliza de una forma que no es normal, o falla toda la máquina, la conexión con la cola del agente se interrumpe y el gestor de colas cierra todas las colas abiertas, incluida la cola SYSTEM.FTE.HA.<agent name> y las conexiones.

Debido a esto, la instancia en espera adquiere el método GET exclusivo y completa el resto de inicialización del agente.

De nuevo, las transferencias en curso se reanudan desde los últimos puntos de comprobación conocidos.

Si se interrumpe una conexión con el gestor de colas

Modalidad de cliente

Un proceso de agente consta de varias hebras. Aparte de las hebras predeterminadas, por ejemplo, una hebra que publica el estado del agente a intervalos regulares, cada solicitud de transferencia se maneja con un conjunto de hebras que finalizan después de que se complete una transferencia.

Muchas de estas hebras se conectan al gestor de colas de agente y colocan y obtienen mensajes. Es posible que alguna de estas conexiones pueda interrumpirse debido a un problema de red o a un gestor de colas que falla. Cuando alguna hebra detecta un problema de interrupción de conexión, la hebra informa a la hebra principal para iniciar la recuperación y finaliza.

A continuación, la hebra principal inicia otra hebra para esperar a que se establezca una conexión con el gestor de colas. Una vez que se ha vuelto conectar, se realiza un intento de adquirir el método GET exclusivo para el agente. Si esto se realiza correctamente, el agente continúa para completar la recuperación y se convierte en la instancia activa. Si el intento de adquirir el método GET exclusivo falla, la instancia se convierte en una instancia en espera.

Modalidad de enlaces

Cuando se conecta en modalidad de enlaces, si un agente pierde la conexión, el proceso del agente finaliza. El controlador de proceso maneja el reinicio del agente. Cuando se reinicia un agente, pasa por el proceso de intentar adquirir el método GET exclusivo para sí mismo.

Si el agente tiene éxito, se convierte en una instancia activa; de lo contrario, el agente se convierte en una instancia en espera.

Aplicación de actualizaciones de nivel de mantenimiento

Los pasos para aplicar el mantenimiento a agentes altamente disponibles son similares a los documentados para gestores de colas multiinstancia. Para obtener más información, consulte Aplicación de actualizaciones de nivel de mantenimiento a gestores de colas de varias instancias en Windows o Aplicación de actualizaciones de nivel de mantenimiento a gestores de colas de varias instancias en AIX, o Aplicación de actualizaciones de nivel de mantenimiento a gestores de colas de varias instancias en Linux.

Debe detener el agente que se ejecuta en la máquina donde se va a aplicar el nivel de mantenimiento, antes de aplicar el mantenimiento. Si está actualizando una instancia activa, para la continuidad de las transferencias, debe migrar tras error de la instancia activa a una instancia en espera.

Una vez que se haya completado la actualización, debe iniciar la instancia del agente, migrar tras error la instancia activa a la instancia actualizada y, después, actualizar la instancia en espera.

Migración de agentes de una versión anterior del producto

Los agentes migrados de versiones de IBM MQ anteriores a IBM MQ 9.1.4 se ejecutan como no altamente disponibles. Puede ejecutarlos en la modalidad de alta disponibilidad siguiendo el procedimiento descrito en Migración de agentes Managed File Transfer de una versión anterior.

Configuración de un registrador de MFT

Cuando Managed File Transfer transfiere archivos, publica información sobre sus acciones en un tema en el gestor de colas de coordinación. El registrador de base de datos es un componente opcional de Managed File Transfer que puede utilizar para copiar esta información en una base de datos para fines de análisis y auditoría.

Existen tres versiones del registrador:

- 

registrador de archivo autónomo
- registrador de base de datos autónomo

- registrador Java Platform, Enterprise Edition (Java EE)

Registradores en IBM i

IBM i

Los registradores de Managed File Transfer no están soportados en la plataforma IBM i.

Registrador de archivo autónomo

ALW

El registrador de archivo autónomo es un proceso Java que se ejecuta en el sistema que aloja el gestor de colas de coordinación, o en un sistema que aloja un gestor de colas con conectividad con el gestor de colas de coordinación. El registrador de archivo autónomo utiliza enlaces de IBM MQ para conectarse al gestor de colas asociado. El registrador autónomo se crea utilizando el mandato **fteCreateLogger**.

Windows Puede ejecutar el registrador de archivo autónomo como un servicio de Windows para asegurarse de que el registrador de archivo siga ejecutándose cuando se desconecte de la sesión de Windows, y se puede configurar para que se inicie automáticamente cuando se reinicie un sistema. Para obtener más información, consulte “Instalación del registrador de archivo autónomo de MFT” en la página 821.

El registrador de archivos autónomo no está soportado en las plataformas siguientes:

- z/OS z/OS
- IBM i IBM i

Registrador de base de datos autónomo

El registrador de base de datos autónomo es una aplicación Java que se instala en un sistema que aloja un gestor de colas y una base de datos. El registrador de base de datos autónomo suele instalarse en el mismo sistema que el gestor de colas de coordinación, pero también se puede instalar en el mismo sistema que cualquier gestor de colas que tenga conectividad con el gestor de colas que tenga conectividad con el gestor de colas de coordinación. El registrador de base de datos autónomo utiliza los enlaces de IBM MQ para conectarse a su gestor de colas asociado, y un controlador JDBC de tipo 2 o de tipo 4 para conectarse a una base de datos Db2 u Oracle. Estos tipos de conexión son necesarios porque el registrador de base de datos autónomo utiliza el soporte XA del gestor de colas para coordinar una transacción global a través del gestor de colas y de la base de datos, protegiendo los datos.

Windows Si está utilizando un sistema Windows , puede ejecutar los registradores autónomos como servicios de Windows para garantizar que los registradores sigan ejecutándose cuando finalice la sesión de Windows. Si desea más información, consulte “Instalar el registrador de base de datos autónomo de MFT” en la página 829 para un registrador de base de datos autónomo.

Registrador de base de datos de Java EE

El registrador de base de datos Java EE se proporciona como un archivo EAR, que se instala en un servidor de aplicaciones. Esto puede resultar más conveniente que utilizar el registrador de base de datos autónomo si tiene un entorno de servidor de aplicaciones Java EE existente, ya que el registrador de base de datos Java EE puede ser gestionado junto con otras aplicaciones empresariales. También puede instalar el registrador de base de datos Java EE en un sistema independiente de los sistemas que alojan el servidor de IBM MQ y la base de datos. El registrador de base de datos Java EE está soportado para su uso con bases de datos Db2 y Oracle. El registrador de base de datos Java EE también da soporte a Oracle Real Application Clusters cuando está instalado en WebSphere Application Server 7.0.

Para obtener instrucciones sobre cómo configurar un registrador, consulte los temas siguientes:

- “Instalación del registrador de archivo autónomo de MFT” en la página 821

- “Instalar el registrador de base de datos autónomo de MFT” en la página 829
- “Instalación del registrador de base de datos Java EE para MFT” en la página 833

Tareas relacionadas

“Utilización de MFT con una base de datos remota” en la página 830
Puede utilizar el registrador de Managed File Transfer para comunicarse con una base de datos en un sistema remoto.

Referencia relacionada

Manejo de errores del registrador de MFT y rechazo de mensajes
Propiedades de configuración del registrador de MFT

ALW Instalación del registrador de archivo autónomo de MFT

El registrador de archivos autónomo es un proceso Java que puede conectarse a un gestor de colas de coordinación utilizando la modalidad de enlaces IBM MQ o la modalidad de cliente. Para definir un registrador de archivo autónomo, utilice el mandato **fteCreateLogger** y siga los pasos de este tema.

Acerca de esta tarea

Para obtener más información sobre el registrador de archivo autónomo, consulte “Configuración de un registrador de MFT” en la página 819. Los pasos de este tema configuran un registrador para conectarse a un gestor de colas de coordinación. Para configuraciones de registrador alternativas, consulte “Configuraciones alternativas para un registrador autónomo de MFT” en la página 832.

El registrador de archivos autónomo no está soportado en las plataformas siguientes:

- z/OS z/OS
- IBM i IBM i

Procedimiento

1. Asegúrese de tener instalado el componente Managed File Transfer Logger. Para obtener más información, consulte Opciones del producto Managed File Transfer.
2. Ejecute el mandato **fteCreateLogger**, especificando el gestor de colas de coordinación y estableciendo el parámetro **-loggerType** en `FILE`, para crear el registrador de archivo autónomo.

   Si desea más información, consulte fteCreateLogger.
3. Opcional: Si desea utilizar un formato personalizado, puede modificar el archivo XML creado por el mandato **fteCreateLogger**. La definición de formato de registro se encuentra en el archivo `FileLoggerFormat.xml`.

   Para obtener más información, consulte “Formato de registrador de archivos autónomo de MFT” en la página 822.
4. Ejecute los mandatos MQSC, proporcionados por el mandato **fteCreateLogger** , en el gestor de colas de coordinación para crear las colas de registrador.
5. Identifique un usuario para ejecutar el proceso de registrador y configure permisos para ese usuario.

   Para obtener más información, consulte “Configuración del acceso de usuario para un registrador de MFT” en la página 828.
6. Opcional: Configure más el registrador de archivo autónomo editando el archivo `logger.properties` creado al ejecutar el mandato **fteCreateLogger** .

   Este archivo es un archivo de propiedades Java que consta de pares de clave-valor. El archivo `logger.properties` se encuentra en el directorio *MQ_DATA_PATH*`/mqft/config/`*coordination_qmgr_name*`/loggers/`*logger_name* . Para obtener más información sobre las propiedades disponibles y sus efectos, consulte Propiedades de configuración del registrador de MFT.
7. Windows

Opcional: Si está utilizando un sistema Windows , ejecute el registrador de archivos autónomo como un servicio Windows .

Ejecute el mandato **fteModifyLogger** con el parámetro **-s**. Si desea más información, consulte fteModifyLogger.

8. Inicie el registrador de archivo autónomo con el mandato **fteStartLogger**.

   Si desea más información, consulte fteStartLogger.

   Si ha realizado el paso anterior y ha utilizado el mandato **fteModifyLogger** con el parámetro **-s** en Windows, el registrador de archivo autónomo arranca como un servicio Windows.
9. Compruebe la salida del registrador. El registrador de archivo autónomo genera dos tipos de salida, datos de auditoría de transferencia de archivos y datos de diagnóstico del registrador.

   Los datos de auditoría de transferencia de archivos se pueden encontrar en *MQ_DATA_PATH*/mqft/logs/*coordination_qmgr_name*/loggers/*logger_name*/logs. Los datos de diagnóstico del registrador se pueden encontrar en *MQ_DATA_PATH*/mqft/logs/*coordination_qmgr_name*/loggers/*logger_name*.
10. Detenga el registrador utilizando el mandato **fteStopLogger** .

    Si desea más información, consulte fteStopLogger.

Resultados

Tareas relacionadas

“Configuración del acceso de usuario para un registrador de MFT” en la página 828
En un entorno de prueba, puede añadir nuevos privilegios necesarios a la cuenta normal de usuario. En un entorno de producción, es recomendable crear un nuevo usuario con los permisos mínimos necesarios para realizar el trabajo.

Referencia relacionada

Propiedades de configuración del registrador de MFT

fteStartLogger (iniciar un registrador de MFT)

fteCreateLogger (crear un archivo MFT o registrador de base de datos)

fteModifyLogger (ejecutar un registrador de MFT como un servicio de Windows)

fteStopLogger (detener un registrador de MFT)

“Formato de registrador de archivos autónomo de MFT” en la página 822
El formato de la información de los mensajes escritos por el registrador de archivos se puede definir en el archivo FileLoggerFormat.xml.

Autorizaciones para el registrador de MFT

ALW *Formato de registrador de archivos autónomo de MFT*

El formato de la información de los mensajes escritos por el registrador de archivos se puede definir en el archivo FileLoggerFormat.xml.

El directorio de configuración del registrador se encuentra en *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name*/loggers/*logger_name*. Al crear un nuevo registrador de archivo, se crea una versión de este archivo que contiene un conjunto predeterminado de definiciones utilizado por el registrador de archivo. Para obtener más información sobre la definición de formato de registro predeterminado, consulte Formato de registro predeterminado del registrador de archivos autónomo MFT.

Si desea especificar su propio formato de registro personalizado, edite el archivo FileLoggerFormat.xml.

Una definición de formato de registro personalizada

Una definición de formato de registro consiste en un conjunto de tipos de mensaje en el que cada tipo de mensaje tiene una definición de formato. Una definición de formato para un tipo de mensaje consta de un conjunto de inserciones proporcionadas en formato XPATH y un separador que se utiliza para separar cada inserción. El orden de las inserciones determina el orden en el que se coloca el contenido en las

líneas generadas para la salida en los archivos de registro. Por ejemplo, ésta es la definición para el tipo de mensaje `callStarted`:

```
<callStarted>
  <format>
    <inserts>
      <insert type="user"   width="19" ignoreNull="false">/transaction/action/
       @time</insert>
      <insert type="user"   width="48" ignoreNull="false">/transaction/@ID</insert>
      <insert type="system" width="6"  ignoreNull="false">type</insert>
      <insert type="user"   width="0"  ignoreNull="false">/transaction/agent/
       @agent</insert>
      <insert type="user"   width="0"  ignoreNull="false">/transaction/agent/@QMgr</insert>
      <insert type="user"   width="0"  ignoreNull="false">/transaction/job/name</insert>
      <insert type="user"   width="0"  ignoreNull="true">/transaction/transferSet/
       call/command/@type</insert>
      <insert type="user"   width="0"  ignoreNull="true">/transaction/transferSet/
      call/command/@name</insert>
      <insert type="system" width="0"  ignoreNull="true">callArguments</insert>
    </inserts>
    <separator></separator>
  </format>
</callStarted>
```

Este formato produce una línea en el archivo de registro como la siguiente:

```
2011-11-25T10:53:04;414d5120514d5f67627468696e6b20206466cf4e20004f02;[CSTR];
  AGENT1;AGENT_QM;Managed Call;executable;echo;call test;
```

Las inserciones proporcionadas en la definición de formato están en el orden en que aparece la información en la línea en el archivo de registro. Para obtener más información sobre el esquema XML que define el formato para el archivo `FileLoggerFormat.xml`, consulte XSD de formato de registrador de archivos autónomo.

Tipos de mensaje

Los agentes FTE graban un rango de diferentes tipos de mensajes en el subtema SYSTEM.FTE/Log. Para obtener más información, consulte SYSTEM.FTE TemaFTE. La definición de archivo de registro puede contener definiciones de formato para estos tipos de mensajes:

```
callCompleted
callStarted
monitorAction
monitorCreate
monitorFired
notAuthorized
scheduleDelete
scheduleExpire
scheduleSkipped
scheduleSubmitInfo
scheduleSubmitTransfer
scheduleSubmitTransferSet
transferStarted
transferCancelled
transferComplete
transferDelete
transferProgress
```

El formato de los mensajes puede variar. La mayoría de tipos de mensaje graban una sola línea en el archivo de registro para cada mensaje de registro consumido del subtema SYSTEM.FTE/Log. Esto lleva al caso simple en el que las direcciones XPATH proporcionadas en la definición de formato de registro se refieren a la raíz del mensaje. Estos son los tipos de mensaje que utilizan este método para grabar salida:

```
callCompleted
callStarted
monitorAction
monitorCreate
monitorFired
notAuthorized
```

```
scheduleDelete
scheduleExpire
scheduleSkipped
scheduleSubmitInfo
scheduleSubmitTransfer
transferStarted
transferCancelled
transferComplete
transferDelete
```

El otro método utilizado para grabar un mensaje de registro utiliza varias líneas para representar los elementos de un conjunto de transferencias dentro de un mensaje de registro. En este caso, se aplica el formato proporcionado a cada elemento del conjunto de transferencias del mensaje de registro. Si desea incluir información que sea específica de cada elemento del conjunto de transferencias, es necesario que el XPATH proporcionado utilice el elemento como la raíz XPATH. Estos son los tipos de mensaje que utilizan este método para grabar salida:

```
scheduleSubmitTransferSet
transferProgress
```

Se graba una línea de salida para cada elemento en el conjunto de transferencias. La información que desee establecer como fija para todos los elementos de un conjunto de transferencias puede seguir utilizando direcciones XPATH relativas a la raíz del mensaje de registro. En el siguiente ejemplo simplificado de definición de formato `transferProgress`, se fija la indicación de fecha y hora y el ID de transferencia. Toda la información relativa a un elemento como su raíz variará para cada línea escrita. En este ejemplo, se graba la información de archivo de origen y destino para cada elemento.

```
<transferProgress>
      <format>
        <inserts>
          <insert type="user"   width="19" ignoreNull="false">/transaction/action/
           @time</insert>
          <insert type="user"   width="48" ignoreNull="false">/transaction/@ID</insert>
          <insert type="system" width="6"  ignoreNull="false">type</insert>
          <insert type="user"   width="3"  ignoreNull="true">status/@resultCode</insert>
          <insert type="user"   width="0"  ignoreNull="false">source/file |
           source/queue</insert>
          <insert type="user"   width="0"  ignoreNull="false">source/file/@size |
           source/queue/@size</insert>
          <insert type="user"   width="5"  ignoreNull="true">source/@type</insert>
          <insert type="user"   width="6"  ignoreNull="true">source/@disposition</insert>
          <insert type="user"   width="0"  ignoreNull="false">destination/file |
            destination/queue</insert>
          <insert type="user"   width="0"  ignoreNull="false">destination/file/@size |
            destination/queue/@size</insert>
          <insert type="user"   width="5"  ignoreNull="true">destination/@type</insert>
          <insert type="user"   width="9"  ignoreNull="true">destination/@exist</insert>
          <insert type="user"   width="0"  ignoreNull="true">status/supplement</insert>
        </inserts>
        <separator></separator>
      </format>
    </transferProgress>
```

Esto produce una entrada de archivo de registro de una o más líneas con este formato:

```
2011-11-25T13:45:16;414d5120514d5f67627468696e6b20206466cf4e20033702;[TPRO];0
  ;/src/test1.file;3575;file;leave ;/dest/test1.file;3575;file;overwrite;;
2011-11-25T13:45:16;414d5120514d5f67627468696e6b20206466cf4e20033702;[TPRO];0
  ;/src/test2.file;3575;file;leave ;/dest/test2.file;3575;file;overwrite;;
```

Formato de inserción

Hay dos tipos de inserción disponibles al definir un formato para un tipo de mensaje: `user` y `system`. El tipo de una inserción se define en el atributo `type` del elemento insert. También se puede personalizar

el diseño de estos dos tipos de inserciones utilizando los atributos **width** e **ignoreNull** del elemento insert. Por ejemplo:

```
<insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
```

En este ejemplo, la inserción toma la información que se encuentra en el mensaje de registro en `/transaction/@ID` y la recorta o rellena a 48 caracteres antes de escribirla en el registro. Si el contenido de `/transaction/@ID` es nulo, se graba la serie null después de rellenarla hasta 48 caracteres porque el atributo `ignoreNull` está establecido en `false`. Si `ignoreNull` se establece en `true`, en lugar de ello se graba la serie vacía, rellenada a 48 caracteres. Establecer width="0 " significa que la anchura de la columna no se recorta, no significa que la anchura se recorta a 0. El atributo `ignoreNull` se puede utilizar de esta forma para detectar en el registro cuando se encuentra un valor nulo cuando no se esperaba. Esto puede ser útil al depurar una nueva definición de archivo de registro.

Inserciones definidas por el usuario

Una inserción de usuario contiene una dirección XPATH para la información que se va a grabar en esa inserción. Esta dirección hace referencia a un fragmento de información que se encuentra en el mensaje de registro FTE. Para obtener más información sobre los formatos de mensaje de registro, consulte:

- Formatos de mensajes de registro de transferencias de archivos
- Formatos de mensajes de registro de transferencia de archivos planificada
- Formato de mensaje de registro de supervisores de MFT

Inserciones definidas por el sistema

Las inserciones definidas por el sistema contienen una palabra clave que hace referencia a una información que no se encuentra en el mensaje de registro o que no es fácil de definir mediante el lenguaje XPATH.

Las inserciones del sistema soportadas son:

- `type` - Graba el tipo del mensaje de registro en un formato corto.
- `callArguments` - Graba el conjunto de argumentos proporcionados a una llamada gestionada en un formato separado por espacios.
- `transferMetaData` - Graba el conjunto de entradas de metadatos definidas para una transferencia en un formato *clave=valor* separado por comas.

En la tabla siguiente se lista el valor de "type" para inserciones definidas por el sistema para cada tipo de mensaje.

Tabla 51. Resumen de tipos de mensaje soportados y las inserciones de sistema "type".

Tipo de mensaje	**Valor de inserción de sistema "type"**
callCompleted	[CCOM]
callStarted	[CSTR]
monitorAction	[MACT]
monitorCreate	[MCRT]
monitorFired	[MFIR]
notAuthorized	[AUTH]
scheduleDelete	[SDEL]

Tabla 51. Resumen de tipos de mensaje soportados y las inserciones de sistema "type". (continuación)	
Tipo de mensaje	**Valor de inserción de sistema "type"**
scheduleExpire	[SEXP]
scheduleSkipped	[SSKP]
scheduleSubmitInfo	[SSIN]
scheduleSubmitTransfer	[SSTR]
scheduleSubmitTransferSet	[SSTS]
transferStarted	[TSTR]
transferCancelled	[TCAN]
transferComplete	[TCOM]
transferDelete	[TDEL]
transferProgress	[TPRO]

Referencia relacionada

Formato de registro predeterminado de registrador de archivos autónomo de MFT
XSD del formato de registrador de archivo autónomo
Tema SYSTEM.FTE
Formatos de mensajes de registro de transferencias de archivos
Formatos de mensajes de registro de transferencia de archivos planificada
Formato de mensaje de registro de supervisores de MFT

ALW *Exclusión de tipos de mensaje del registrador de archivo autónomo de MFT*

Si desea excluir un determinado tipo de mensaje de la salida de registrador de archivo, puede utilizar elementos de tipo de mensaje vacíos.

Ejemplo

Por ejemplo, la siguiente definición de formato impide que el registrador de archivo genere como salida mensajes `transferProgress`.

```
<?xml version="1.0" encoding="UTF-8"?>
<logFormatDefinition  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="1.00"
 xsi:noNamespaceSchemaLocation="FileLoggerFormat.xsd">
  <messageTypes>
    <transferProgress></transferProgress>
  </messageTypes>
</logFormatDefinition>
```

ALW *Definición de formatos personalizados para el registrador de archivos autónomo de MFT*

Es posible definir un subconjunto de tipos de mensaje personalizados en una definición de formato de registro para reducir la cantidad de configuración necesaria para personalizar el formato de archivo de registro.

Acerca de esta tarea

Si no se incluye un elemento `messageTypes` en el archivo `FileLoggerFormat.xml`, el formato para ese tipo de mensaje utiliza el formato predeterminado. Sólo necesita especificar los formatos que desea que sean diferentes del valor predeterminado.

Ejemplo

En este ejemplo, la definición de formato sustituye el formato predeterminado para el tipo de mensaje `transferStarted` con esta versión reducida que sólo genera como salida el usuario que ha iniciado la transferencia. Todos los demás tipos de mensaje utilizan el formato predeterminado porque no están incluidos en esta definición de formato de registro:

```
<?xml version="1.0" encoding="UTF-8"?>
<logFormatDefinition  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="1.00"
 xsi:noNamespaceSchemaLocation="FileLoggerFormat.xsd">
  <messageTypes>
    <transferStarted>
      <format>
        <inserts>
          <insert type="user"   width="19" ignoreNull="false">/transaction/action/
           @time</insert>
          <insert type="user"   width="48" ignoreNull="false">/transaction/@ID</insert>
          <insert type="system" width="6"  ignoreNull="false">type</insert>
          <insert type="user"   width="0"  ignoreNull="true">/transaction/originator/
           userID</insert>
        </inserts>
        <separator>;</separator>
      </format>
    </transferStarted>
  </messageTypes>
</logFormatDefinition>
```

Referencia relacionada

Formato de registro predeterminado de registrador de archivos autónomo de MFT

XSD del formato de registrador de archivo autónomo

ALW *Reducción de mensajes duplicados en el registrador de archivo autónomo de MFT*

Pueden aparecer mensajes de registro duplicados en el registro del registrador de archivo autónomo. Utilizando el archivo `logger.properties` puede ajustar el registrador de archivos autónomo y reducir el número de duplicados.

Duplicar mensajes en el registro de registrador de archivo

En el caso de una anomalía, es posible que se grabe un mensaje de registro en el registro del registrador de archivo autónomo sin el consumo del mensaje de registro del SYSTEM.FTE/Log# tema que se confirma en IBM MQ. Si sucede esto, cuando el registrador de archivo autónomo se reinicie, recuperará el mismo mensaje una segunda vez y lo grabará de nuevo en el archivo de registro. Planifique manejar la posibilidad de estos duplicados manualmente al buscar los archivos de registro o automáticamente al procesarlos. Para ayudarle en la detección de duplicados, el registrador de archivo autónomo genera el mensaje siguiente en el archivo de registro cuando se inicia:

```
BFGDB0054I: The file logger has successfully started
```

Los duplicados siempre se producen alrededor de la hora de inicio del registrador de archivo autónomo, porque ése es el momento en que se procesa el último mensaje leído antes de que fallara la instancia anterior. Si se sabe cuándo se ha iniciado la instancia nueva, se puede detectar si se deben esperar duplicados y si es necesario manejarlos o no.

Reducir el número de duplicados

El registrador de archivo autónomo agrupa los mensajes de registro que procesa en transacciones para mejorar el rendimiento. Este tamaño de lote es el número máximo de mensajes duplicados que puede que necesite ver en caso de anomalía. Para reducir el número de duplicados, puede ajustar la siguiente propiedad en el archivo `logger.properties`:

```
wmqfte.max.transaction.messages
```

Por ejemplo, estableciendo este valor en 1 el número máximo de mensajes duplicados se reduce a 1. Tenga en cuenta que la modificación de este valor tiene un efecto sobre el rendimiento del registrador de archivos autónomo, por lo que es necesario realizar pruebas exhaustivas para asegurarse de que esto no afecta negativamente al sistema.

El archivo `logger.properties` se encuentra en el directorio *MQ_DATA_PATH*`/mqft/config/`*coordination_qmgr_name*`/loggers/`*logger_name*. Para obtener más información sobre las propiedades disponibles y sus efectos, consulte Propiedades de configuración del registrador de MFT

ALW *Configuración del acceso de usuario para un registrador de MFT*

En un entorno de prueba, puede añadir nuevos privilegios necesarios a la cuenta normal de usuario. En un entorno de producción, es recomendable crear un nuevo usuario con los permisos mínimos necesarios para realizar el trabajo.

Acerca de esta tarea

Debe instalar el registrador de archivo autónomo y IBM MQ en un solo sistema. Configure los permisos del usuario del modo siguiente:

Procedimiento

1. Asegúrese de que el usuario tiene permiso para leer y, cuando sea necesario, para ejecutar los archivos instalados como parte de la instalación de Managed File Transfer.
2. Asegúrese de que el usuario tiene permiso para crear y escribir en cualquier archivo del directorio `logs` que se encuentre en el directorio de configuración. Este directorio se utiliza para un registro de sucesos y, si es necesario, para archivos de rastreo de diagnóstico y FFDC (First Failure Data Capture - Captura de datos en primer error).
3. Asegúrese de que el usuario tiene su propio grupo y que tampoco está en ningún grupo con una amplia variedad de permisos sobre el gestor de colas de coordinación. El usuario no debe estar en el grupo mqm. En determinadas plataformas, al grupo staff se le concede también automáticamente acceso del gestor de colas; el usuario del registrador de archivo autónomo no debe estar en el grupo staff. Puede ver registros de autorización para el propio gestor de colas y para objetos que este contenga utilizando IBM MQ Explorer. Pulse con el botón derecho del ratón en el objeto y seleccione **Autorizaciones sobre objeto > Gestionar registros de autorización**. En la línea de mandatos, puede utilizar los mandatos dspmqaut (autorización de visualización) o dmpmqaut (autorización de volcado).
4. Utilice la ventana **Gestionar registros de autorización** en IBM MQ Explorer o el mandato setmqaut (otorgar o revocar autorización) para añadir autorizaciones para el propio grupo del usuario (en AIX, las autorizaciones de IBM MQ están asociadas a grupos únicamente, no a usuarios individuales). Las autorizaciones necesarias son las siguientes:
   - Connect e Inquire en el gestor de colas (las bibliotecas IBM MQ Java necesitan el permiso Inquire para realizar operaciones).
   - Permiso Subscribe en el tema SYSTEM.FTE.
   - Permiso Put en la cola SYSTEM.FTE.LOG.RJCT.*nombre_registrador*.
   - Permiso Get en la cola SYSTEM.FTE.LOG.CMD.*nombre_registrador*.

   Los nombres de cola de rechazados y de mandatos especificados son los nombres predeterminados. Si eligió nombres de cola distintos al configurar las colas del registrador de archivo autónomo, añada los permisos a dichos nombres de cola en su lugar.

Instalar el registrador de base de datos autónomo de MFT

Siga estos pasos para instalar y configurar el registrador de base de datos autónomo.

Acerca de esta tarea

Importante: Los registradores de Managed File Transfer no están soportados en la plataforma IBM i.

Para obtener más información sobre el registrador de base de datos autónomo, consulte "Configuración de un registrador de MFT" en la página 819.

Nota: No puede ejecutar más de un registrador de base de datos (autónomo o Java EE) para el mismo esquema en una base de datos simultáneamente. Si intenta hacerlo, se producirán conflictos al intentar grabar datos del registro de transferencias en la base de datos.

Procedimiento

1. Instale el software utilizando la documentación de la base de datos.

   Si el soporte de JDBC es un componente opcional de la base de datos, deberá instalar este componente.
2. Ejecute el mandato **fteCreateLogger** estableciendo el parámetro **-loggerType** en DATABASE para crear el registrador de base de datos autónomo. Si desea más información, consulte fteCreateLogger.

   El nombre de esquema predeterminado es FTELOG. Si utiliza un nombre de esquema distinto de FTELOG, debe editar el archivo SQL proporcionado adecuado a la base de datos, `ftelog_tables_db2.sql` o `ftelog_tables_oracle.sql`, para reflejar este nombre de esquema antes de continuar con el paso siguiente. Para obtener más información, consulte wmqfte.database.schema en Propiedades de configuración del registrador de MFT.
3. Cree las tablas de base de datos necesarias utilizando las herramientas de la base de datos.

   Multi En Multiplatforms, los archivos `ftelog_tables_db2.sql` y `ftelog_tables_oracle.sql` contienen mandatos SQL que puede ejecutar para crear las tablas.

   z/OS En z/OS, el archivo que hay que ejecutar depende de la versión de Db2 for z/OS que se está utilizando:

   - Para Db2 for z/OS 9.0 y anterior, ejecute el archivo `ftelog_tables_zos.sql` para crear las tablas. Este archivo crea las tablas empleando un tipo de datos INTEGER para los campos que indican el tamaño de los archivos transferidos y el ID de tabla asociada a cada transferencia.
   - Para Db2 for z/OS 9.1 y posterior, ejecute el archivo `ftelog_tables_zos_bigint.sql` para crear las tablas. Este archivo crea las tablas empleando un tipo de datos BIGINT para los campos que indican el tamaño de los archivos transferidos y el ID de tabla asociada a cada transferencia.
4. Ejecute los mandatos MQSC, proporcionados por el mandato **fteCreateLogger**, en el gestor de colas de mandatos de registrador para crear las colas del registrador. El registrador de base de datos autónomo utiliza dos colas en el gestor de colas de coordinación. La primera cola es una cola de mandatos donde se colocan los mensajes para controlar la operación del registrador de base de datos autónomo. El nombre predeterminado de esta cola de mandatos es SYSTEM.FTE.LOG.CMD.*nombre_registrador*. La segunda cola es una cola de rechazados. Dado que el registrador de base de datos autónomo nunca descarta mensajes de registro, si el registrador encuentra un mensaje que no puede manejar, lo coloca en la cola de rechazados para examinarlo y para un posible reproceso. No es recomendable utilizar la cola de mensajes no entregados del gestor de colas para este fin, porque los mensajes rechazados no tienen una cabecera DLH y porque los mensajes rechazados no deben combinarse con mensajes transferidos a la cola de mensajes no entregados por otras razones. El nombre predeterminado de la cola de rechazados es SYSTEM.FTE.LOG.RJCT.*nombre_registrador*. Estas dos colas se definen en los archivos de script MQSC generados por el mandato **fteCreateLogger**.
5. Elegir un usuario y configurar permisos

6. Opcional: Puede configurar adicionalmente el registrador de base de datos autónomo editando el archivo `logger.properties` creado por el mandato **fteCreateLogger** en el paso "2" en la página 829. Este archivo es un archivo de propiedades Java que consta de pares de clave-valor. El archivo `logger.properties` se encuentra en el directorio *MQ_DATA_PATH*`/mqft/config/`*coordination_qmgr_name*`/loggers/`*logger_name*. Para obtener más información sobre las propiedades disponibles y sus efectos, consulte Propiedades de configuración del registrador de MFT.
7. Windows
   Opcional: Si utiliza un sistema Windows, puede ejecutar el registrador de base de datos autónomo como un servicio de Windows. Ejecute el mandato **fteModifyLogger** con el parámetro **-s**. Si desea más información, consulte fteModifyLogger.
8. Opcional: Si la base de datos que se utiliza es Oracle o se va a conectar a una base de datos Db2 de forma remota, deberá especificar un nombre de usuario y una contraseña que el registrador utilizará para autenticarse con el servidor de bases de datos. Este nombre de usuario y contraseña se especifican en un archivo de credenciales que se ajusta al formato definido por el esquema `MQMFTCredentials.xsd`. Si desea más información, consulte Formato de archivo de credenciales MFT. Después de crear el archivo de credenciales, debe especificar la ubicación del archivo de credenciales en el archivo `logger.properties` utilizando la propiedad `wmqfte.database.credentials.file`.
9. Inicie el registrador de base de datos autónomo utilizando el mandato **fteStartLogger**. De forma predeterminada, el registrador de base de datos autónomo se ejecuta en segundo plano y el registrador de base de datos autónomo coloca la salida en un archivo del directorio `logs`. Si desea ejecutar el registrador de base de datos autónomo en primer plano y generar la salida en la consola, así como en el archivo de registro, añada el parámetro **-F** al mandato **fteStartLogger**.

   Si ha realizado el paso anterior y ha utilizado el mandato **fteModifyLogger** con el parámetro **-s** en Windows, el registrador de base de datos autónomo se inicia como un servicio de Windows.

Tareas relacionadas

"Configuración del acceso de usuario para un registrador de base de datos autónomo de MFT" en la página 831
En un entorno de prueba, puede añadir nuevos privilegios necesarios a la cuenta normal de usuario. En un entorno de producción, es recomendable crear un nuevo usuario con los permisos mínimos necesarios para realizar el trabajo.

Referencia relacionada

Propiedades de configuración del registrador de MFT
fteStartLogger
fteModifyLogger
Autorizaciones para el registrador de MFT

Utilización de MFT con una base de datos remota

Puede utilizar el registrador de Managed File Transfer para comunicarse con una base de datos en un sistema remoto.

Acerca de esta tarea

Si tiene una base de datos instalada en una máquina distinta de la máquina en la que ha instalado Managed File Transfer, realice los pasos siguientes. Los pasos se aplican a Db2 y Oracle a menos que se indique lo contrario.

Procedimiento

1. Instale un cliente de base de datos en el sistema en el que ha instalado Managed File Transfer.
2. Añada el servidor de base de datos remoto a la configuración del cliente de base de datos local. Esta actualización de la configuración es necesaria para que Managed File Transfer y IBM MQ accedan correctamente a la base de datos.

3. Especifique las nuevas propiedades en el archivo `logger.properties` para conectarse a la base de datos utilizando el archivo de credenciales: **wmfte.database.credentials.file**.

   Nota: Deprecated Las versiones anteriores de Managed File Transfer utilizaban las propiedades **wmqfte.oracle.user** o **wmqfte.database.user** y **wmqfte.oracle.password** o **wmqfte.database.password**. Estas propiedades están ahora obsoletas. En su lugar, utilice **wmfte.database.credentials.file**.
4. **Sólo Oracle:** Para permitir una conexión remota a la base de datos, cambie la stanza XAResourceManager del archivo `qm.ini` del gestor de colas de coordinación a lo siguiente (asegurándose de cambiar el nombre de la base de datos, el nombre de usuario y la contraseña de usuario para que coincidan con su propia información):

   `Oracle_XA+Acc=P/ftelog/qgw783jhT+SesTm=35+DB=FTEAUDIT1`**`+SqlNet=FTEAUDIT1`**`+threads=false`,
   el cambio se resalta en negrita.
5. **Solo Oracle:** especifique un host y un puerto en el archivo `logger.properties`, utilizando las propiedades **wmqfte.oracle.host** y **wmqfte.oracle.port**. Los valores predeterminados para el host y el puerto permiten trabajan con un cliente de base de datos local, por lo que si ha trabajado anteriormente con una base de datos local, es posible que no haya establecido estos valores.

Referencia relacionada

Propiedades de configuración del registrador de MFT

Configuración del acceso de usuario para un registrador de base de datos autónomo de MFT

En un entorno de prueba, puede añadir nuevos privilegios necesarios a la cuenta normal de usuario. En un entorno de producción, es recomendable crear un nuevo usuario con los permisos mínimos necesarios para realizar el trabajo.

Acerca de esta tarea

El número y tipo de cuentas de usuario que necesita para ejecutar el registrador de base de datos autónomo depende del número de sistemas que utilice. Puede instalar el registrador de base de datos autónomo, IBM MQ y la base de datos en un solo sistema o en dos sistemas. El registrador de base de datos autónomo debe estar en el mismo sistema que IBM MQ. Los componentes pueden instalarse en las siguientes topologías:

Registrador de base de datos autónomos, IBM MQ y la base de datos en el mismo sistema
: Puede definir un único sistema operativo para que sea utilizado por todos los componentes. Ésta es una configuración adecuada para el registrador de base de datos autónomo. El registrador de base de datos autónomo utiliza la modalidad de enlaces para conectarse a IBM MQ y una conexión nativa para conectarse a la base de datos.

El registrador de base de datos autónomo y IBM MQ en un sistema y la base de datos en un sistema distinto
: Creará dos usuarios para esta configuración: un usuario de sistema operativo en el sistema donde se ejecuta el registrador de base de datos autónomo y un usuario de sistema operativo con acceso remoto a la base de datos en el servidor de bases de datos. Ésta es una configuración adecuada para el registrador de base de datos autónomo cuando utiliza una base de datos remota. El registrador de base de datos autónomo utiliza la modalidad de enlaces para conectarse a IBM MQ y una conexión de cliente para acceder a la base de datos.

Como ejemplo, el resto de estas instrucciones presuponen que el usuario se llama `ftelog`, aunque puede utilizarse cualquier otro nombre. Configure los permisos del usuario del modo siguiente:

Procedimiento

1. Asegúrese de que el usuario tiene permiso para leer y, cuando sea necesario, para ejecutar los archivos instalados como parte de la instalación de Herramientas remotas y documentación de Managed File Transfer.

2. Asegúrese de que el usuario tiene permiso para crear y escribir en cualquier archivo del directorio `logs` (en el directorio de configuración). Este directorio se utiliza para un registro de sucesos y, si es necesario, para el rastreo de diagnóstico y archivos FFDC.
3. Asegúrese de que el usuario tiene su propio grupo y tampoco está en ningún grupo con una amplia variedad de permisos sobre el gestor de colas de coordinación. El usuario no debe estar en el grupo mqm. En determinadas plataformas, al grupo staff se le concede también automáticamente acceso del gestor de colas; el usuario del registrador de base de datos autónomo no debe estar en el grupo staff. Puede ver registros de autorización para el propio gestor de colas y para objetos que este contenga utilizando IBM MQ Explorer. Pulse con el botón derecho del ratón en el objeto y seleccione **Autorizaciones sobre objeto > Gestionar registros de autorización**. En la línea de mandatos, puede utilizar los mandatos dspmqaut (autorización de visualización) o dmpmqaut (autorización de volcado).
4. Utilice la ventana **Gestionar registros de autorización** en IBM MQ Explorer o el mandato setmqaut (otorgar o revocar autorización) para añadir autorizaciones para el propio grupo del usuario (en AIX, las autorizaciones de IBM MQ están asociadas a grupos únicamente, no a usuarios individuales). Las autorizaciones necesarias son las siguientes:
   - Connect e Inquire en el gestor de colas (las bibliotecas IBM MQ Java necesitan el permiso Inquire para realizar operaciones).
   - Permiso Subscribe en el tema SYSTEM.FTE.
   - Permiso Put en la cola SYSTEM.FTE.LOG.RJCT.*nombre_registrador*.
   - Permiso Get en la cola SYSTEM.FTE.LOG.CMD.*nombre_registrador*.

   Los nombres de cola de rechazados y de mandatos especificados son los nombres predeterminados. Si eligió nombres de cola distintos al configurar las colas del registrador de base de datos autónomo, añada los permisos a dichos nombres de cola en su lugar.
5. Realice la configuración de usuario que sea específica de la base de datos que esté utilizando.
   - Si la base de datos es Db2, lleve a cabo los pasos siguientes:

     Hay varios mecanismos para gestionar usuarios de bases de datos con Db2. Estas instrucciones se aplican al esquema predeterminado basado en los usuarios del sistema operativo.

     – Asegúrese de que el usuario `ftelog` no esté en ningún grupo de administración de Db2 (por ejemplo, db2iadm1, db2fadm1 o dasadm1)
     – Asigne al usuario permiso para conectarse a la base de datos y permiso para seleccionar, insertar y actualizar las tablas que ha creado como parte del Paso 2: crear las tablas de base de datos necesarias.
   - Si la base de datos es Oracle, lleve a cabo los pasos siguientes:
     – Asegúrese de que el usuario `ftelog` no esté en ningún grupo de administración de Oracle (por ejemplo, ora_dba en Windows o dba en AIX and Linux)
     – Asigne al usuario permiso para conectarse a la base de datos y permiso para seleccionar, insertar y actualizar las tablas que ha creado como parte del Paso 2: crear las tablas de base de datos necesarias.

Configuraciones alternativas para un registrador autónomo de MFT

Normalmente, un registrador autónomo de Managed File Transfer, ya sea un archivo o un tipo de base de datos, está en el mismo sistema que el gestor de colas de coordinación y se conecta al gestor de colas de coordinación en modalidad de enlaces de IBM MQ. Sin embargo, también se puede instalar en el mismo sistema que cualquier gestor de colas que tenga conectividad con el gestor de colas de coordinación. El registrador autónomo recibe mensajes mediante una suscripción, que el registrador autónomo crea automáticamente. Esta es la configuración que se describe en las instrucciones de instalación.

No obstante, si tiene consideraciones específicas del sitio, puede configurar un registrador autónomo para que reciba mensajes de otras dos maneras, controladas por la propiedad wmqfte.message.source.type. Esta propiedad se describe en Propiedades de configuración del registrador de MFT.

Suscripción administrativa

De forma predeterminada, un registrador autónomo crea su propia suscripción al tema SYSTEM.FTE/Log/#, utilizando las opciones de suscripción duradera predeterminadas y una suscripción gestionada (es decir, el gestor de colas controla la cola de reserva que se utiliza para guardar los mensajes antes de que se pasen a la aplicación). Si se necesitan otras opciones en la suscripción o la cola, puede crear usted mismo una suscripción, establecer las opciones que necesita y configurar el registrador autónomo para que utilice esa suscripción. Recuerde que debe añadir el permiso para que el registrador autónomo utilice la suscripción que cree.

Un ejemplo de utilización de esta configuración es particionar el espacio de registro mediante dos suscripciones de comodín, para enviar registros de los agentes cuyo nombre empieza por FINANCE a una base de datos y registros de los agentes que empiezan por ACCOUNTING a otra base de datos. Este tipo de configuración requiere dos instancias de registrador autónomas, cada una con su propio archivo `logger.properties` que hace referencia a la suscripción necesaria y a su propia cola de mandatos y cola de rechazos.

Para recopilar mensajes de registro únicamente de los agentes cuyos nombres empiezan por ACCOUNTING, cree un objeto de suscripción en el gestor de colas de coordinación con una serie de tema de SYSTEM.FTE/Log/ACCOUNTING*. Establezca el valor **Uso de comodín** en **Comodín a nivel de carácter**. También debe añadir entradas al archivo `logger.properties` para el registrador. Por ejemplo, si crea un objeto de suscripción denominado ACCOUNTING.LOGS con estos valores, añada las entradas siguientes al archivo `logger.properties`:

```
wmqfte.message.source.type=administrative subscription
wmqfte.message.source.name=ACCOUNTING.LOGS
```

El registrador autónomo maneja solamente los mensajes de registro que empiezan por la serie de tema de SYSTEM.FTE/Log/. Puede especificar una serie de tema más restrictiva, pero no puede especificar una serie menos restrictiva. Si especifica una serie menos restrictiva por error, todas las publicaciones que están relacionadas con una serie de tema que no sea SYSTEM.FTE/Log/ irán a la cola de rechazados y el registrador autónomo generará el mensaje de error BFGDB0002E. Este mensaje de error implica que hay un problema en la configuración del registrador autónomo.

Cola

La topología típica es aquella en que el registrador autónomo se ejecuta en el mismo sistema que el gestor de colas de coordinación. Si ello no fuera posible, puede crear una suscripción en el gestor de colas de coordinación utilizando una cola en otro gestor de colas como destino de suscripción (utilizando una definición de cola remota o utilizando la propiedad DESTQMGR de la suscripción). El registrador puede entonces ejecutarse en el sistema que aloja el segundo gestor de colas y leer mensajes de la cola. Para garantizar la integridad transaccional, el registrador autónomo siempre debe conectarse al gestor de colas en modalidad de enlaces. Debe definir la cola de rechazados y la cola de mandatos en el mismo gestor de colas al que se conecta el registrador autónomo. Los gestores de colas deben estar en IBM WebSphere MQ 7.5 o posterior.

Por ejemplo, para recopilar mensajes de registro que se están colocando en la cola USER.QUEUE mediante una suscripción, añada estas entradas al archivo `logger.properties`:

```
wmqfte.message.source.type=queue
wmqfte.message.source.name=USER.QUEUE
```

Instalación del registrador de base de datos Java EE para MFT

Siga estas instrucciones para instalar y configurar el registrador de base de datos JEE para utilizarlo con Managed File Transfer.

Acerca de esta tarea

Para obtener más información sobre el registrador de base de datos Java EE, consulte el tema "Configuración de un registrador de MFT" en la página 819.

Nota: No puede ejecutar un registrador de base de datos Java EE al mismo tiempo que un registrador autónomo, a menos que estos registradores estén utilizando instancias separadas de la base de datos.

Procedimiento

1. Antes de instalar el registrador de base de datos Java EE, debe preparar el entorno. Siga las instrucciones del tema "Preparación para instalar el registrador de base de datos Java EE para MFT" en la página 834.
2. Instale el registrador de base de datos Java EE en un servidor de aplicaciones compatible con Java Platform, Enterprise Edition (Java EE) o Jakarta EE .

   Para obtener instrucciones, consulte "Instalación del registrador de base de datos Java EE para MFT con WebSphere Application Server traditional 9.0" en la página 837

Tareas relacionadas

"Preparación para instalar el registrador de base de datos Java EE para MFT" en la página 834
Siga estas instrucciones para preparar el entorno de Managed File Transfer antes de instalar el registrador de base de datos Java EE.

"Instalación del registrador de base de datos Java EE para MFT con WebSphere Application Server traditional 9.0" en la página 837
Siga estas instrucciones para instalar y configurar el registrador de base de datos Java Platform, Enterprise Edition (Java EE) para Managed File Transfer con WebSphere Application Server traditional 9.0.

"Configuración del acceso de usuario para el registrador de base de datos Java EE para MFT" en la página 842
Cuando se configure el registrador de base de datos Java Platform, Enterprise Edition (Java EE) para Managed File Transfer, necesita cuentas de usuario para acceder a IBM MQ, la base de datos y el sistema operativo. El número de usuarios del sistema operativo depende del número de sistemas que esté utilizando para alojar estos componentes.

"Migración del registrador de base de datos autónomo al registro de base de datos Java EE para MFT" en la página 843
Puede migrar desde el registrador de base de datos autónomo al registrador de base de datos Java EE. Debe detener el registrador de base de datos autónomo e instalar el registrador de base de datos JEE. Para evitar perder o duplicar entradas de registro, debe detener la publicación de mensajes en SYSTEM.FTE antes de detener el registrador de base de datos autónomo y reiniciarlo después de instalar el registrador de base de datos de Java EE . Haga una copia de seguridad de su base de datos antes de realizar la migración.

Referencia relacionada

Autorizaciones para el registrador de MFT

Preparación para instalar el registrador de base de datos Java EE para MFT

Siga estas instrucciones para preparar el entorno de Managed File Transfer antes de instalar el registrador de base de datos Java EE.

Acerca de esta tarea

Para obtener más información sobre el registrador de base de datos Java EE, consulte el tema "Configuración de un registrador de MFT" en la página 819.

Procedimiento

1. Instale el software utilizando la documentación de la base de datos.

Si el soporte de JDBC es un componente opcional de la base de datos, deberá instalar este componente.

2. Cree una base de datos utilizando las herramientas proporcionadas por la base de datos. La base de datos debe tener un espacio de tabla y un tamaño de página de agrupación de almacenamiento intermedio de al menos 8K.

   El nombre de esquema predeterminado es FTELOG. Si utiliza un nombre de esquema distinto de FTELOG, debe editar el archivo SQL proporcionado adecuado a la base de datos, `ftelog_tables_db2.sql` o `ftelog_tables_oracle.sql`, para reflejarlo antes de continuar con el paso siguiente.

   Nota: Los archivos `ftelog_tables_db2.sql` y `ftelog_tables_oracle.sql` están en la vía de acceso de archivos `<MQ-installation-path>/mqft/sql`

3. Cree las tablas de base de datos necesarias utilizando las herramientas de la base de datos.

   Multi En Multiplatforms, los archivos `ftelog_tables_db2.sql` y `ftelog_tables_oracle.sql` contienen mandatos SQL que puede ejecutar para crear las tablas.

   z/OS En z/OS, el archivo que hay que ejecutar depende de la versión de Db2 for z/OS que se está utilizando:

   - Para Db2 for z/OS 9.0 y anterior, ejecute el archivo `ftelog_tables_zos.sql` para crear las tablas. Este archivo crea las tablas empleando un tipo de datos INTEGER para los campos que indican el tamaño de los archivos transferidos y el ID de tabla asociada a cada transferencia.
   - Para Db2 for z/OS 9.1 y posterior, ejecute el archivo `ftelog_tables_zos_bigint.sql` para crear las tablas. Este archivo crea las tablas empleando un tipo de datos BIGINT para los campos que indican el tamaño de los archivos transferidos y el ID de tabla asociada a cada transferencia.

4. Si ha cambiado el nombre de esquema de FTELOG a otro de su elección, debe cambiar el nombre de esquema en el archivo EAR. Para obtener más información, consulte "Cambio del nombre de esquema en el registrador de base de datos Java EE para MFT" en la página 835.
5. Cree una cola de rechazados en IBM MQ.

   Dado que el registrador nunca descarta mensajes de registro, si el registrador encuentra un mensaje que no puede manejar, lo coloca en la cola de rechazados para examinarlo y para un posible reproceso. No utilice la cola de mensajes no entregados del gestor de colas para este fin, porque los mensajes rechazados no tienen una cabecera DLH y porque los mensajes rechazados no deben combinarse con mensajes transferidos a la cola de mensajes no entregados por otras razones. El mandato **`fteCreateLogger`** crea una cola de rechazados. El nombre predeterminado para esta cola de rechazados es SYSTEM.FTE.LOG.RJCT.*nombre_registrador*
6. Siga las instrucciones del tema "Configuración del acceso de usuario para el registrador de base de datos Java EE para MFT" en la página 842.

Qué hacer a continuación

Instale el registrador de base de datos Java EE en un servidor de aplicaciones compatible con Java EE o Jakarta EE . Utilice las instrucciones de "Instalación del registrador de base de datos Java EE para MFT con WebSphere Application Server traditional 9.0" en la página 837

Cambio del nombre de esquema en el registrador de base de datos Java EE para MFT

El registrador de base de datos Java Platform, Enterprise Edition (Java EE) puede utilizar una base de datos que tenga un nombre de esquema no predeterminado, Debe cambiar el nombre de esquema en el archivo EAR del registrador de base de datos Java EE.

Acerca de esta tarea

Para cambiar el nombre del esquema que utiliza el registrador de base de datos Java EE, complete los pasos siguientes:

Procedimiento

1. Extraiga el archivo JAR JPA del archivo EAR utilizando el siguiente mandato:

```
jar -xvf ear_file lib/jpa_file
```

   donde:

   - *ear_file* es `com.ibm.wmqfte.databaselogger.jee.oracle.ear` o `com.ibm.wmqfte.databaselogger.jee.ear` en función de si está utilizando Db2 u Oracle.
   - *jpa_file* es `com.ibm.wmqfte.web.jpa.oracle.jar` o `com.ibm.wmqfte.web.jpa.jar` en función de si está utilizando Db2 u Oracle.

2. Extraiga el archivo `persistence.xml` del archivo JPA JAR utilizando el mandato siguiente:

```
jar -xvf lib/jpa_file META_INF/persistence.xml
```

   donde:

   - *jpa_file* es `com.ibm.wmqfte.web.jpa.oracle.jar` o `com.ibm.wmqfte.web.jpa.jar` en función de si está utilizando Db2 u Oracle.

3. Edite el archivo `persistence.xml` para cambiar la línea siguiente:

```
<property name="openjpa.jdbc.Schema" value="schema_name" />
```

   donde

   - *nombre_esquema* es el nombre de esquema que desea utilizar.

4. Actualice JAR JPA con el archivo `persistence.xml` modificado utilizando el mandato siguiente:

```
jar -uvf lib/jpa_file META_INF/persistence.xml
```

   donde:

   - *jpa_file* es `com.ibm.wmqfte.web.jpa.oracle.jar` o `com.ibm.wmqfte.web.jpa.jar` en función de si está utilizando Db2 u Oracle.

5. Actualice el archivo EAR con el archivo JAR JPA modificado, utilizando el siguiente mandato:

```
jar -uvf ear_file lib/jpa_file
```

   donde:

   - *ear_file* es `com.ibm.wmqfte.databaselogger.jee.oracle.ear` o `com.ibm.wmqfte.databaselogger.jee.ear` en función de si está utilizando Db2 u Oracle.
   - *jpa_file* es `com.ibm.wmqfte.web.jpa.oracle.jar` o `com.ibm.wmqfte.web.jpa.jar` en función de si está utilizando Db2 u Oracle.

Qué hacer a continuación

Utilice el archivo EAR modificado para instalar el registrador de base de datos Java EE.

Tareas relacionadas

“Instalación del registrador de base de datos Java EE para MFT con WebSphere Application Server traditional 9.0” en la página 837

Siga estas instrucciones para instalar y configurar el registrador de base de datos Java Platform, Enterprise Edition (Java EE) para Managed File Transfer con WebSphere Application Server traditional 9.0.

Establecer la vía de acceso de biblioteca nativa en WebSphere Application Server traditional 9.0

Si despliega la aplicación de registrador de base de datos de Java Platform, Enterprise Edition (Java EE) en WebSphere Application Server traditional 9.0 y desea utilizar conexiones de modalidad de enlaces entre la aplicación y IBM MQ, debe configurar el proveedor de mensajería de IBM MQ con la ubicación de las bibliotecas nativas de IBM MQ en el sistema.

Acerca de esta tarea

Si no establece la vía de acceso de biblioteca nativa en el servidor de aplicaciones, puede recibir el siguiente mensaje de error en el registro de salida del sistema de WebSphere Application Server traditional 9.0:

```
A connection could not be made to WebSphere MQ for the following reason:
  CC=2;RC=2495;AMQ8568: The native JNI library 'mqjbnd' was not found. [3=mqjbnd]
```

Utilice la consola administrativa de WebSphere Application Server traditional 9.0 para completar los siguientes pasos:

Procedimiento

1. En el panel de navegación, expanda **Recursos** > **JMS** > **Proveedores de JMS**.
2. Seleccione el proveedor de mensajería de IBM MQ que esté en el ámbito correcto para la fábrica de conexiones o la especificación de activación que crea la conexión de modalidad de enlaces.

   Nota: La información de vía de acceso nativa en el ámbito de `Server` se utiliza en lugar de la información de vía de acceso nativa en ámbitos superiores, y la información de vía de acceso nativa en el ámbito de `Node` se utiliza en lugar de la información de vía de acceso nativa en el ámbito de `Cell` .
3. Bajo Propiedades generales, en el campo **Vía de acceso de biblioteca nativa**, escriba el nombre completo del directorio que contiene las bibliotecas nativas de IBM MQ.

   Por ejemplo, en Linux escriba `/opt/mqm/java/lib`. Escriba un solo nombre de directorio.
4. Pulse **Aceptar**.

   Una vez establecida la vía de acceso, debe guardar los cambios en la configuración maestra para que los cambios entren en vigor.
5. Reinicie el servidor de aplicaciones para renovar la configuración.
6. Necesario: Reinicie el servidor de aplicaciones por segunda vez para cargar las bibliotecas.

Tareas relacionadas

“Instalación del registrador de base de datos Java EE para MFT con WebSphere Application Server traditional 9.0” en la página 837

Siga estas instrucciones para instalar y configurar el registrador de base de datos Java Platform, Enterprise Edition (Java EE) para Managed File Transfer con WebSphere Application Server traditional 9.0.

Instalación del registrador de base de datos Java EE para MFT con WebSphere Application Server traditional 9.0

Siga estas instrucciones para instalar y configurar el registrador de base de datos Java Platform, Enterprise Edition (Java EE) para Managed File Transfer con WebSphere Application Server traditional 9.0.

Antes de empezar

Antes de instalar la aplicación de registrador de base de datos JEE, siga las instrucciones de los temas “Preparación para instalar el registrador de base de datos Java EE para MFT” en la página 834 y “Establecer la vía de acceso de biblioteca nativa en WebSphere Application Server traditional 9.0” en la página 837.

Acerca de esta tarea

Para obtener más información sobre el registrador de base de datos Java EE, consulte "Configuración de un registrador de MFT" en la página 819.

Procedimiento

1. Configure el proveedor de JDBC de XA:
   a) Seleccione **Recursos** > **JDBC** > **Proveedores JDBC** en la navegación de la consola administrativa de WebSphere Application Server traditional 9.0.
   b) Cree un proveedor de JDBC utilizando el asistente de la consola pulsando **Nuevo**.
   c) En el Paso 1 del asistente, seleccione la base de datos que está utilizando en la lista **Tipo de base de datos**, y el tipo de proveedor asociado en la lista **Tipo de proveedor**. En la lista **Tipo de implementación**, seleccione **Origen de datos XA**. Pulse **Siguiente**.

      z/OS Puede eliminar una referencia a `db2jcc_license_cisuz.jar` y debe cambiar `db2jcc.jar` por `db2jcc4.jar`, es decir, la versión del archivo jar que se entrega con la versión más reciente de Db2 o la versión local.
   d) En el Paso 2 del asistente, compruebe que la ubicación del directorio de los archivos JAR de la base de datos requerida están establecidos correctamente. Pulse **Siguiente**.
   e) Pulse **Finalizar** en la pagina de resumen para crear el proveedor de JDBC.
2. Cree alias de autenticación. Creará un alias para el origen de datos y otro para IBM MQ:
   a) Seleccione **Seguridad** > **Seguridad global** en la navegación de la consola de administración de WebSphere Application Server traditional 9.0.
   b) En el encabezado **Autenticación**, expanda **Autenticación de Java y servicio de autorización**.
   c) Pulse **Datos de autenticación de J2C**. Se abrirá la página del alias de autenticación .
   d) Cree un alias de autenticación para el origen de datos:
      i) Pulse **Nuevo**.
      ii) Entre los detalles para **Alias**, **ID de usuario**, **Contraseña** y **Descripción**. Los detalles que se entran en los campos **ID de usuario** y **Contraseña** deben coincidir con los detalles que ha entrado al crear el usuario de base de datos. Para obtener más información, consulte "Configuración del acceso de usuario para el registrador de base de datos Java EE para MFT" en la página 842.
      iii) Pulse **Aceptar**.
   e) Cree un alias de autenticación para IBM MQ:
      i) Pulse **Nuevo**.
      ii) Entre los detalles para **Alias**, **ID de usuario**, **Contraseña** y **Descripción**. Los detalles que se entran en los campos **ID de usuario** y **Contraseña** deben coincidir con los valores de usuario y contraseña de la instalación de IBM MQ.
      iii) Pulse **Aceptar**.
3. Cree un origen de datos:
   a) Seleccione **Recursos** > **JDBC** > **Orígenes de datos** en la navegación de la consola administrativa de WebSphere Application Server traditional 9.0.
   b) Seleccione la lista desplegable **Ámbito** y cambie el ámbito al valor correspondiente. Por ejemplo, `Node=yourNode, Server=yourServer`.
   c) Cree un origen de datos utilizando el asistente de la consola pulsando **Nuevo**.
   d) En el Paso 1 del asistente, en el campo **Nombre de origen de datos**, especifique `wmqfte-database` y en el campo **Nombre JNDI**, especifique `jdbc/wmqfte-database`. Pulse **Siguiente**.
   e) En el Paso 2 del asistente, utilice la lista desplegable **Seleccionar un proveedor de JDBC existente** para seleccionar el proveedor de JDBC creado en los pasos anteriores. Pulse **Siguiente**.
   f) **Db2:** En el Paso 3 del asistente, en el campo **Tipo de controlador**, escriba 4.

g) **Db2:** Especifique los detalles en los campos **Nombre de base de datos**, **Nombre de servidor** y **Número de puerto**, y pulse **Siguiente**.

Oracle: Entre el URL de conexión en el campo **URL** y elija el ayudante de almacén de datos correcto en el campo **Nombre de clase de ayudante de almacén de datos**.

Oracle RAC: Al conectarse a un Oracle Real Application Cluster, el URL de conexión debe incluir la información de host necesaria para conectarse a todas las instancias disponibles de la base de datos.

h) En el paso 4 del asistente, seleccione el nombre del alias de autenticación de origen de datos que definió en el paso 2d en la lista **alias de autenticación para recuperación XA**. Seleccione el mismo nombre en las listas **Alias de autenticación gestionado por componente** y **Alias de autenticación gestionado por contenedor**.

i) Pulse **Finalizar** en la página de resumen para crear el origen de datos.

4. Opcional: Compruebe la configuración del origen de datos:
   a) Seleccione **Recursos** > **JDBC** > **Orígenes de datos** en la navegación de la consola administrativa de WebSphere Application Server traditional 9.0.
   b) Pulse el botón **Probar conexión**.
5. Cree un tema.
   a) Desde la navegación de la consola de administración de WebSphere Application Server traditional 9.0, pulse **Recursos** > **JMS** > **Temas**.
   b) Seleccione la lista desplegable **Ámbito** y cambie el ámbito al valor correspondiente. Por ejemplo, `Node=yourNode, Server=yourServer`.
   c) Pulse **Nuevo**.
   d) Pulse **Proveedor de mensajería de IBM MQ**.
   e) En el panel **Administración** de la página de propiedades del tema, elija valores exclusivos para los campos **Nombre** y **Nombre JNDI**, a los que haré referencia más tarde en la configuración.
   f) En el panel **Tema de IBM MQ**, especifique `SYSTEM.FTE/Log/#` en el campo **Nombre de tema**.
6. Cree una especificación de activación:
   a) Desde la navegación de la consola de administración de WebSphere Application Server traditional 9.0, pulse **Recursos** > **JMS** > **Especificaciones de activación**.
   b) Seleccione la lista desplegable **Ámbito** y cambie el ámbito al valor correspondiente. Por ejemplo, `Node=yourNode, Server=yourServer`.
   c) Pulse **Nuevo**.
   d) Pulse **Proveedor de mensajería de IBM MQ**.
   e) En el Paso 1 del asistente, seleccione valores exclusivos para los campos **Nombre** y **Nombre JNDI**, a los que hará referencia más tarde en la configuración.
   f) En el Paso 1.1, escriba el nombre JNDI del tema que configuró en el paso 5 del campo **Nombre JNDI de destino**.
   g) En la lista **Tipo de destino**, seleccione **Tema**.
   h) En el Paso 1.2 del asistente, seleccione **Suscripción duradera**. Escriba `SYSTEM.FTE.DATABASELOGGER.AUTO` en el campo **Nombre de suscripción**.
   i) En el Paso 2 del asistente, seleccione **Escriba toda la información requerida en este asistente**.
   j) En el Paso 2.1, escriba el nombre del gestor de colas en el campo **Nombre del gestor de colas o grupo compartiendo cola**.
   k) En el Paso 2.2, seleccione el método de transporte que desee en la lista **Transporte**. Si selecciona **Enlaces**, no necesitará ninguna otra información. Si selecciona **Cliente** o **Enlaces y después cliente**, entre los detalles para **Nombre de host**, **Puerto** y **Canal conexión de servidor**.

l) Opcional: Pulse **Probar conexión** para confirmar que el gestor de colas está presente. Sin embargo, está previsto que reciba `NOT_AUTHORIZED` hasta que haga referencia al alias de autenticación en el paso 6n.

m) Pulse **Guardar**.

n) Pulse el nombre de la especificación de activación que ha creado. En la sección **Propiedades generales** de la pestaña **Configuración**, desplácese hasta el panel **Avanzada** y escriba un nombre exclusivo para identificar la conexión de IBM MQ en el campo **ID de cliente**. Debe completar este paso o IBM MQ rechazará la conexión con el código de error `JMSCC0101`.

o) Si eligió **Cliente** como método de transporte, desplácese hasta el panel **Configuración de seguridad** y seleccione el alias de autenticación definido en el paso 8 de la lista **Alias de autenticación**.

p) Haga clic en **Aplicar**.

q) En la sección **Propiedades adicionales** de la pestaña **Configuración**, pulse **Propiedades avanzadas**. En la sección **Consumidor de conexión** del panel **Propiedades avanzadas**, escriba 1 en el campo **Número máximo de sesiones de servidor**.

Nota: Asegúrese de que ha completado este paso antes de continuar. Si no lo hace, puede que el registrador no funcione correctamente.

r) En la sección **Propiedades adicionales** de la pestaña **Configuración**, pulse **Propiedades avanzadas**. Establezca el valor de **Detener punto final si falla la entrega del mensaje** en un mínimo de 1.

Si el valor de la propiedad **`_numberOfFailedAttemptsBeforeReject`** se establece en más de 1 (consulte 9j para obtener más información), establezca **Detener punto final si la entrega de mensajes falla** en al menos el valor de la propiedad **`_numberOfFailedAttemptsBeforeReject`** . Esto impide que el punto final se detenga cuando se recibe un mensaje que no se puede procesar (por ejemplo, un mensaje de registro de transferencias con formato incorrecto). Si desea más información, consulte Manejo y rechazo de errores del registrador MFT.

7. Cree una fábrica de conexiones de cola.

a) Desde la navegación de la consola de administración de WebSphere Application Server traditional 9.0, pulse **Recursos** > **JMS** > **Fábricas de conexiones de cola**.

b) Seleccione la lista desplegable **Ámbito** y cambie el ámbito al valor correspondiente. Por ejemplo, `Node=yourNode, Server=yourServer`.

c) Pulse **Nuevo**.

d) Pulse **Proveedor de mensajería de IBM MQ**.

e) En el Paso 1 del asistente, seleccione valores exclusivos para los campos **Nombre** y **Nombre JNDI**, a los que hará referencia más tarde en la configuración.

f) En el Paso 2, seleccione **Escriba toda la información requerida en este asistente**.

g) En el Paso 2.1, escriba el nombre del gestor de colas en el campo **Nombre del gestor de colas o grupo compartiendo cola**.

h) En el Paso 2.2, seleccione el método de transporte que desee en la lista **Transporte**. Si selecciona **Enlaces**, no necesitará ninguna otra información. Si selecciona **Cliente** o **Enlaces y después cliente**, entre los detalles para **Nombre de host**, **Puerto** y **Canal conexión de servidor**.

i) Opcional: Pulse **Probar conexión** para confirmar que el gestor de colas está presente. Sin embargo, está previsto que reciba `NOT_AUTHORIZED` hasta que haga referencia al alias de autenticación en el paso 7h.

j) Si ha seleccionado **Cliente** o **Enlaces y después cliente** como método de transporte, pulse el nombre de la fábrica de conexión de cola que acaba de crear. Desplácese hasta el panel **Configuración de seguridad** de la pestaña **Configuración** y seleccione el alias de autenticación que ha definido en el paso 2e en las listas **Alias de autenticación para recuperación XA** y **Alias de autenticación gestionado por contenedor**.

8. Cree una cola de rechazados en WebSphere Application Server:

a) Desde la navegación de la consola de administración de WebSphere Application Server traditional 9.0, pulse **Recursos** > **JMS** > **Colas**.

b) Seleccione la lista desplegable **Ámbito** y cambie el ámbito al valor correspondiente. Por ejemplo, `Node=yourNode, Server=yourServer`.

c) Pulse **Nuevo**.

d) Pulse **Proveedor de mensajería de IBM MQ**.

e) Seleccione valores exclusivos para los campos **Nombre** y **Nombre JNDI**, a los que hará referencia más tarde en la configuración.

f) Especifique `SYSTEM.FTE.LOG.RJCT.`*`logger_name`* en el campo **Nombre de cola** . Asegúrese de haber creado esta cola en el gestor de colas de coordinación.

g) Escriba el nombre del gestor de cola en el campo **Nombre de gestor de cola**.

h) Pulse **Aceptar**.

9. Instale la aplicación del registrador de base de datos de JEE:

a) En la consola administrativa de WebSphere Application Server traditional 9.0, seleccione **Aplicaciones** > **Nueva aplicación**.

b) Seleccione la lista desplegable **Ámbito** y cambie el ámbito al valor correspondiente. Por ejemplo, `Node=yourNode, Server=yourServer`.

c) En la lista de opciones, seleccione **Nueva aplicación empresarial**.

d) En la página **Preparación para la instalación de la aplicación**, seleccione el archivo `com.ibm.wmqfte.databaselogger.jee.ear` o el archivo `com.ibm.wmqfte.databaselogger.jee.oracle.ear` del directorio *`MQ_INSTALLATION_PATH`*`/mqft/web` de la instalación de Managed File Transfer Service y pulse **Siguiente**.

e) En la siguiente pantalla, seleccione **Detallado** para mostrar todas las opciones de instalación y parámetros, y pulse **Siguiente**.

f) Pulse **Siguiente** a través de los pasos de asistente 1-4 para aceptar los valores predeterminados.

g) En el paso 5 del asistente, **Enlazar escuchas para beans impulsados por mensajes** , desplácese hasta la sección **Enlaces de escuchas**. Pulse **Especificación de activación**.

Escriba los valores requeridos en los siguientes campos:

Nombre JNDI de recurso de destino
El nombre JNDI que especificó cuando creó una especificación de activación en le paso 6d.

Nombre NDI de destino
El nombre JNDI que especificó cuando creó un tema en el paso 5d.

Pulse **Siguiente**.

h) En el paso 6 del asistente, **Correlacionar referencias de recursos con recursos**, entre los detalles en el campo **Nombre JNDI de recurso de destino**. Este nombre es el nombre JNDI que especificó para la fábrica de conexiones de la cola de rechazados en el paso 7c. Pulse **Siguiente**.

i) En el paso 7 del asistente, **Correlacionar referencias de entradas del entorno de recursos con recursos**, entre los detalles en el campo **Nombre JNDI de recurso de destino**. Este nombre es el nombre JNDI de la cola de rechazados que ha creó en el paso 8d. Pulse **Siguiente**.

j) En el paso 8 del asistente, **Correlacionar entradas de entorno para módulos EJB**, acepte el valor predeterminado de 1. Pulse **Siguiente**.

Oracle RAC: Al conectarse a un Oracle Real Application Cluster se debe establecer el valor de la propiedad _numberOfFailedAttemptsBeforeReject en **como mínimo** 2. Esta propiedad determina el número de veces que el registrador intenta procesar un mensaje de auditoría después de que se produzca una anomalía. En un caso de migración tras error de base de datos, es probable que se produzca al menos una anomalía. Para evitar mover un mensaje innecesariamente a la cola de rechazados, aumentar este valor permite realizar un segundo intento, lo que normalmente tiene un resultado satisfactorio ya que se establece una conexión con la nueva instancia de base de datos. Si descubre durante las pruebas que se siguen moviendo mensajes a la cola de rechazados

durante la migración tras error de su instancia de base de datos, aumente aún más este valor: la sincronización del momento de la conmutación entre las instancias puede provocar más de una anomalía para el mismo mensaje. No obstante, tenga en cuenta que aumentar este valor afecta a todos los casos de anomalías (por ejemplo, un mensaje en formato incorrecto) y no sólo a la sustitución por anomalía de base de datos, de modo que aumente el valor con cuidado para evitar reintentos innecesarios.

k) En el paso 9 del asistente, **Metadatos para módulos**, pulse **Siguiente**.

l) En el paso 10 del asistente, **Resumen**, pulse **Finalizar**.

10. Ahora puede iniciar la aplicación desde la consola administrativa de WebSphere Application Server traditional 9.0:
    a) Seleccione **Aplicaciones** > **Tipos de aplicación** > **Aplicaciones de empresa WebSphere** desde la navegación de la consola.
    b) Seleccione el recuadro de selección para la aplicación empresarial del **Registrador de anotaciones** de la tabla de colección y pulse **Iniciar**.

Configuración del acceso de usuario para el registrador de base de datos Java EE para MFT

Cuando se configure el registrador de base de datos Java Platform, Enterprise Edition (Java EE) para Managed File Transfer, necesita cuentas de usuario para acceder a IBM MQ, la base de datos y el sistema operativo. El número de usuarios del sistema operativo depende del número de sistemas que esté utilizando para alojar estos componentes.

Acerca de esta tarea

El número y tipo de cuentas de usuarios que necesita para ejecutar el registrador de base de datos Java EE dependen del número de sistemas que utilice. Se necesitan cuentas de usuario para acceder a los tres entornos siguientes:

- Sistema operativo local
- IBM MQ
- Base de datos

Puede instalar el registrador de base de datos de JEE, IBM MQ y la base de datos en un único sistema o en varios sistemas. Los componentes pueden instalarse en las siguientes topologías de ejemplo:

El registrador de base de datos Java EE, IBM MQ y la base de datos todos en el mismo sistema
Puede definir un único sistema operativo para que sea utilizado por todos los componentes. El registrador utiliza la modalidad de enlaces para conectarse a IBM MQ y una conexión nativa para conectarse a la base de datos.

El registrador de base de datos Java EE y IBM MQ en un sistema, la base de datos en un sistema separado
Creará dos usuarios para esta configuración: un usuario de sistema operativo en el sistema donde se ejecuta el registrador y un usuario de sistema operativo con acceso remoto a la base de datos en el servidor de bases de datos. El registrador utiliza la modalidad de enlaces para conectarse a IBM MQ y una conexión de cliente para acceder a la base de datos.

El registrador de base de datos Java EE en un sistema, IBM MQ en otro sistema, la base de datos en un sistema adicional
Creará tres usuarios para esta configuración: un usuario del sistema operativo para iniciar el servidor de aplicaciones, un usuario de IBM MQ para acceder a las colas y temas utilizados y un usuario del servidor de bases de datos para acceder e insertar en las tablas de la base de datos. El registrador utiliza la modalidad de cliente para acceder a IBM MQ y una conexión de cliente para acceder a la base de datos.

Como ejemplo, el resto de estas instrucciones presuponen que el usuario se llama `ftelog`, pero puede utilizar cualquier nombre de usuario, nuevo o existente. Configure los permisos del usuario del modo siguiente:

Procedimiento

1. Asegúrese de que el usuario del sistema operativo tiene su propio grupo y tampoco está en ningún grupo con una amplia variedad de permisos sobre el gestor de colas de coordinación. El usuario no debe estar en el grupo mqm. En determinadas plataformas, al grupo staff se le concede también automáticamente acceso del gestor de colas; el usuario del registrador no debe estar en el grupo staff. Puede ver registros de autorización para el propio gestor de colas y para objetos que este contenga utilizando IBM MQ Explorer. Pulse con el botón derecho del ratón en el objeto y seleccione **Autorizaciones sobre objeto > Gestionar registros de autorización**. En la línea de mandatos, puede utilizar los mandatos dspmqaut (autorización de visualización) o dmpmqaut (autorización de volcado).
2. Utilice la ventana **Gestionar registros de autorización** en IBM MQ Explorer o el mandato setmqaut (otorgar o revocar autorización) para añadir autorizaciones para el propio grupo del usuario de IBM MQ (en AIX, las autorizaciones de IBM MQ están asociadas solo a grupos, no a usuarios individuales). Las autorizaciones necesarias son las siguientes:
   - CONNECT e INQUIRE en el gestor de colas (las bibliotecas de IBM MQ Java requieren el permiso INQUIRE para funcionar).
   - Permiso SUBSCRIBE en el tema SYSTEM.FTE.
   - Permiso PUT en la cola SYSTEM.FTE.LOG.RJCT.*nombre_registrador*.

   Los nombres de cola de rechazados y de mandatos especificados son los nombres predeterminados. Si eligió nombres de cola distintos al configurar las colas del registrador, añada los permisos a dichos nombres de cola en su lugar.
3. Realice la configuración del usuario de base de datos específica de la base de datos que esté utilizando.
   - Si la base de datos es Db2, lleve a cabo los pasos siguientes:

     Nota: Hay varios mecanismos para gestionar usuarios de bases de datos con Db2. Estas instrucciones se aplican al esquema predeterminado basado en los usuarios del sistema operativo.

     – Asegúrese de que el usuario de `ftelog` no esté en ningún grupo de administración de Db2 (por ejemplo, db2iadm1, db2fadm1 o dasadm1).

     – Asigne el permiso de usuario para conectarse a la base de datos y el permiso para seleccionar, insertar y actualizar en las tablas que ha creado como parte del Paso 2: crear las tablas de base de datos necesarias.
   - Si la base de datos es Oracle, lleve a cabo los pasos siguientes:

     – Asegúrese de que el usuario de `ftelog` no esté en ningún grupo de administración de Oracle (por ejemplo, ora_dba en Windows o dba en AIX and Linux).

     – Asigne el permiso de usuario para conectarse a la base de datos y el permiso para seleccionar, insertar y actualizar en las tablas que ha creado como parte del Paso 2: crear las tablas de base de datos necesarias.

Migración del registrador de base de datos autónomo al registro de base de datos Java EE para MFT

Puede migrar desde el registrador de base de datos autónomo al registrador de base de datos Java EE. Debe detener el registrador de base de datos autónomo e instalar el registrador de base de datos JEE. Para evitar perder o duplicar entradas de registro, debe detener la publicación de mensajes en SYSTEM.FTE antes de detener el registrador de base de datos autónomo y reiniciarlo después de instalar el registrador de base de datos de Java EE . Haga una copia de seguridad de su base de datos antes de realizar la migración.

Acerca de esta tarea

Procedimiento

1. Antes de detener la base de datos, ejecute el siguiente mandato MQSC en el gestor de colas de coordinación: `ALTER QM PSMODE(COMPAT)`

   Esto detiene la publicación de mensajes en el tema SYSTEM.FTE/Log. Espere hasta que el registrador haya procesado todos los mensajes en su suscripción. De forma predeterminada, esta suscripción se denomina SYSTEM.FTE.LOGGER.AUTO.
2. Detenga el registrador de base de datos utilizando el mandato **fteStopLogger**.
3. Haga una copia de seguridad de la base de datos utilizando las herramientas suministradas por el software de base de datos.
4. Suprima la suscripción que pertenece al registrador de base de datos autónomo.

   De forma predeterminada, esta suscripción se denomina SYSTEM.FTE.LOGGER.AUTO.
5. Si el esquema de base de datos se encuentra en una versión anterior, debe migrarlo a cada nivel posterior en orden. Por ejemplo, si el esquema de base de datos se encuentra en la V7.0.1 y realiza la migración a V7.0.4, debe migrar el esquema de V7.0.1 a V7.0.2, luego de V7.0.2 a V7.0.3 y luego de V7.0.3 a V7.0.4. Migre el esquema de base de datos de la versión *old* a la versión *new*, donde *old* y *new* son variables que describen una versión de esquema, realizando una de las acciones siguientes para cada versión del esquema que debe migrar:

   - z/OS Si la base de datos es Db2 en z/OS y se va a migrar entre esquemas V7.0.2 y V7.0.3 o entre esquemas V7.0.3 y V7.0.4, hay que crear un nuevo esquema de base de datos y copiar en él los datos existentes. Para obtener más información, consulte la documentación de Db2.
   - Si la base de datos no es Db2 o se ha creado la base de datos con un tamaño de página superior a 8K, se puede migrar el esquema del mismo modo que en las otras versiones, siguiendo los pasos siguientes.
   - Si va a realizar la migración entre tablas de base de datos en cualquier otra circunstancia, realice los pasos siguientes:

     a. Elija el archivo que sea adecuado para la plataforma de base de datos y tenga un nombre que incluya la serie *old-new*. Este archivo se encuentra en el directorio *MQ_INSTALLATION_PATH*`/mqft/sql` de la instalación de Herramientas remotas y documentación.

     b. Si ha realizado modificaciones en el esquema inicial, revise el archivo de migración para asegurar que el archivo será compatible con su base de datos modificada.

     c. Ejecute el archivo SQL contra su base de datos.
6. Instale el archivo EAR del registrador de base de datos Java EE.
7. Despliegue el registrador de base de datos Java EE. Para obtener más información, consulte “Instalación del registrador de base de datos Java EE para MFT” en la página 833.
8. Ejecute el mandato MQSC siguiente en el gestor de colas de coordinación: `ALTER QMGR PSMODE(ENABLED)`

   Esto habilita la publicación de mensajes en el tema SYSTEM.FTE/Log.

Resultados

Configurar el puente Connect:Direct

Configure el puente Connect:Direct para transferir archivos entre una red de Managed File Transfer y una red de Connect:Direct. Los componentes del puente Connect:Direct son un nodo Connect:Direct y un agente de Managed File Transfer que está dedicado a comunicarse con dicho nodo. Este agente se denomina el agente de puente Connect:Direct.

Antes de empezar

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 AIX
- 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.

Para obtener más detalles de las versiones de sistema operativo soportadas para el puente de Connect:Direct, consulte la página web Requisitos del sistema para IBM MQ.

Acerca de esta tarea

Un agente de puente de Connect:Direct es un agente de Managed File Transfer que se dedica a la comunicación con un nodo de Connect:Direct.

De forma predeterminada, el agente de puente Connect:Direct utiliza el protocolo TCP/IP para conectarse al nodo Connect:Direct. Si desea una conexión segura entre el agente de puente Connect:Direct y el nodo Connect:Direct, puede utilizar el protocolo SSL o el protocolo TLS.

Procedimiento

1. Elija los sistemas operativos para el agente de puente y el nodo Connect:Direct:
   a) Elija un sistema que ejecute Windows o Linux en x86-64 para instalar el agente de puente de Connect:Direct en.
   b) Elija un sistema operativo que esté soportado por Connect:Direct para Windows o Connect:Direct para UNIX para instalar el nodo de puente Connect:Direct.
2. Elija y configure un nodo Connect:Direct.
   Para poder seguir con estas instrucciones, debe tener instalado un nodo Connect:Direct.
   a) Elija un nodo Connect:Direct para que el agente de Managed File Transfer se comunique con él.
   b) Compruebe el mapa de red para el nodo Connect:Direct elegido. Si el mapa de red contiene entradas de nodos remotos que ejecutan en un sistema operativo Windows, hay que asegurarse de que dichas entradas especifiquen que los nodos ejecutan en Windows.

      Si el nodo Connect:Direct que ha seleccionado para el puente de Connect:Direct se está ejecutando en Windows, utilice el Peticionario de Connect:Direct para editar la correlación de red. Asegúrese de que el campo **Sistema operativo** de los nodos remotos que ejecutan en Windows está establecido a **Windows**.
3. Cree y configure un agente de puente Connect:Direct.
   a) Cree un agente de puente Connect:Direct utilizando el mandato **fteCreateCDAgent**.
      - Debe proporcionar un valor para el parámetro **cdNode**. Este parámetro especifica el nombre que el agente utiliza para el nodo Connect:Direct que forma parte del puente Connect:Direct. Utilice el nombre del nodo Connect:Direct que ha elegido en la sección anterior.

- Proporcione valores para los parámetros **cdNodeHost** y **cdNodePort**, que definen el nodo Connect:Direct con el que se comunica el agente.

  Si proporciona un valor para el parámetro **cdNodeHost**, se utiliza el nombre de host o la dirección IP del sistema local. Si no proporciona un valor para el parámetro **cdNodePort**, se utiliza el valor 1363.
- De forma opcional, utilice la información de fteCreateAgent para determinar si tiene que especificar un valor para el parámetro **cdTmpDir**.

b) Correlacione las credenciales de usuario utilizadas por Managed File Transfer con las credenciales de usuario en un nodo Connect:Direct. Puede correlacionar credenciales utilizando uno de los métodos siguientes:

- Cree un archivo `ConnectDirectCredentials.xml` para definir la información de correlación de credenciales. Para obtener más información, consulte "Correlación de credenciales de Connect:Direct utilizando el archivo ConnectDirectCredentials.xml" en la página 847.
- Escriba una salida de usuario para realizar la correlación de credenciales para el puente Connect:Direct. Para obtener más información, consulte "Correlación de credenciales en Connect:Direct mediante clases de salida" en la página 849.

4. Configure el archivo `ConnectDirectNodeProperties.xml` para incluir información sobre los nodos Connect:Direct remotos.

   Debe haber creado un agente de puente Connect:Direct antes de seguir estas instrucciones.

   Edite la plantilla `ConnectDirectNodeProperties.xml` en el directorio de configuración del agente de puente de Connect:Direct. Para cada nodo o grupo de nodos Connect:Direct para los que desee definir información, realice los pasos siguientes:

   a) Dentro del elemento nodeProperties, cree un elemento node.

   b) Añada un atributo name al elemento node. Especifique el valor de este atributo como un patrón que coincida con el nombre de uno o más nodos Connect:Direct remotos.

   c) Opcional: Añada un atributo pattern al elemento node que especifique qué tipo de patrón es el valor del atributo name. Los valores válidos son `regex` y `wildcard`. La opción predeterminada es `wildcard`.

   d) Añada un atributo type al elemento node que especifique el sistema operativo en el que se ejecutan los nodos Connect:Direct remotos especificados por el atributo name.

   Los siguientes valores son válidos:

   - `Windows` - el nodo ejecuta en Windows
   - `UNIX`: el nodo se ejecuta en AIX and Linux
   - z/OS `z/OS`, `zos`, `os/390`, u `os390` - el nodo ejecuta en z/OS

   El valor de este atributo no es sensible a mayúsculas y minúsculas. Las transferencias a nodos remotos en otros sistemas operativos no están soportadas por el puente Connect:Direct.

   Si desea más información, consulte Formato de archivo de propiedades de nodo Connect:Direct.

5. Configure una conexión segura entre el agente de puente Connect:Direct y el nodo Connect:Direct. Si desea un ejemplo de cómo hacerlo, consulte Configuración de SSL o TLS entre el agente del puente Connect:Direct y el nodo de Connect:Direct.

Tareas relacionadas

Resolución de problemas con el puente Connect:Direct

Configuración de SSL o TLS entre el agente de puente Connect:Direct y el nodo Connect:Direct

Transferencia de un archivo a un nodo Connect:Direct

Transferencia de un archivo desde un nodo Connect:Direct

z/OS Transferencia de varios archivos desde un nodo Connect:Direct

Referencia relacionada

El puente Connect:Direct

Correlación de credenciales en Connect:Direct

Correlacione credenciales de usuario de Managed File Transfer con credenciales de usuario en un nodo Connect:Direct utilizando la función de correlación de credenciales predeterminada del agente de puente de Connect:Direct 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.

Tareas relacionadas

"Correlación de credenciales de Connect:Direct utilizando el archivo ConnectDirectCredentials.xml" en la página 847

Correlacione credenciales de usuario de Managed File Transfer con credenciales de usuario en nodos Connect:Direct utilizando la función de correlación de credenciales predeterminada del agente de puente Connect:Direct. Managed File Transfer proporciona un archivo XML que puede editar para incluir información sobre las credenciales.

"Correlación de credenciales en Connect:Direct mediante clases de salida" en la página 849

Si no desea utilizar la función de correlación de credenciales predeterminada del agente de puente Connect:Direct, puede correlacionar las credenciales de usuario de Managed File Transfer con las credenciales de usuario en un nodo Connect:Direct escribiendo su propia salida de usuario. Si configura sus propias salidas de usuario de correlación de credenciales, se inhabilita la función de correlación de credenciales predeterminada.

Referencia relacionada

Interfaz CDCredentialExit.java

Formato del archivo de credenciales Connect:Direct

Correlación de credenciales de Connect:Direct utilizando el archivo ConnectDirectCredentials.xml

Correlacione credenciales de usuario de Managed File Transfer con credenciales de usuario en nodos Connect:Direct utilizando la función de correlación de credenciales predeterminada del agente de puente Connect:Direct. Managed File Transfer proporciona un archivo XML que puede editar para incluir información sobre las credenciales.

Acerca de esta tarea

Después de crear un agente de puente Connect:Direct utilizando el mandato **fteCreateCDAgent**, debe crearse manualmente un archivo `ConnectDirectCredentials.xml`. Antes de poder utilizar un agente de puente Connect:Direct, debe editar este archivo para incluir información de host, de usuario y de credenciales. Si desea más información, consulte Formato de archivo de credenciales Connect:Direct. De forma predeterminada, este archivo se carga desde el directorio inicial del usuario actual, `/home/fteuser/ConnectDirectCredentials.xml` por ejemplo. Para utilizar otra ubicación, especifíquela utilizando el elemento `<credentialsFile>` en el archivo `ConnectDirectNodeProperties.xml`.

Procedimiento

1. Asegúrese de que el atributo `name` del elemento `<tns:pnode name="Connect:Direct node host" pattern="wildcard">` contiene el valor del nombre del nodo Connect:Direct al que se conecta el agente de puente Connect:Direct . Este valor debe ser el mismo que el especificado en el parámetro **fteCreateCDAgent -cdNode**.

   El valor del atributo `pattern` puede ser `wildcard` o `regex`. Si no se especifica este atributo, el valor predeterminado es `wildcard`.
2. Inserte el ID de usuario y la información de credenciales en el archivo como elementos hijo de `<tns:pnode>`.

   Puede insertar una o más instancias del siguiente elemento `<tns:user>` en el archivo:

```
<tns:user name="name"
          pattern="pattern"
          ignorecase="ignorecase"
          cdUserId="cdUserId"
```

```
          cdPassword="cdPassword"
          pnodeUserId="pnodeUserId"
          pnodePassword="pnodePassword">
</tns:user>
```

donde:

- *name* es un patrón para comparar con el ID de usuario de MQMD asociado a la solicitud de transferencia de MFT.
- *pattern* especifica si el patrón especificado en el atributo name es una expresión comodín o una expresión regular Java. El valor del atributo pattern puede ser wildcard o regex. Si no se especifica este atributo, el valor predeterminado es wildcard.
- *ignorecase* especifica si hay que tratar el patrón especificado por el atributo name como sensible a mayúsculas y minúsculas. Si no se especifica este atributo, el valor predeterminado es true.
- *cdUserId* es el ID de usuario que utiliza el agente de puente Connect:Direct para conectarse al nodo Connect:Direct que especifica el atributo name del elemento <tns:pnode>. Si es posible, asegúrese de que *cdUserId* sea un ID de usuario administrador de Connect:Direct. Si *cdUserId* no puede ser un administrador de Connect:Direct, asegúrese de que el ID de usuario tiene las autorizaciones funcionales siguientes en el nodo de puente Connect:Direct:

  – En un nodo Windows establezca las autorizaciones siguientes. Este ejemplo se ha formateado con retornos de carro para mejorar la legibilidad:

  ```
  View Processes in the  value: yes
   TCQ
  Issue the copy         value: yes
   receive, copy send,
   run job, and run task
   Process statements
  Issue the submit       value: yes
   Process statement
  Monitor, submit,       value: all
   change,and delete all
   Processes
  Access Process         value: all
   statistics
  Use the trace tool or  value: yes
   issue traceon and
   traceoff commands
  Override Process       value: yes
   options such as file
   attributes and remote
   node ID
  ```

  – Para un nodo AIX o Linux, establezca los siguientes parámetros en el archivo userfile.cfg:

  ```
  pstmt.copy             value: y
  pstmt.upload           value: y
  pstmt.download         value: y
  pstmt.runjob           value: y
  pstmt.runtask          value: y
  cmd.submit             value: y
  pstmt.submit           value: y
  cmd.chgproc            value: y
  cmd.delproc            value: y
  cmd.flsproc            value: y
  cmd.selproc            value: a
  cmd.selstats           value: a
  cmd.trace              value: y
  snode.ovrd             value: y
  ```

- *cdPassword* es la contraseña asociada con el ID de usuario que especifica el atributo cdUserId.
- Puede especificar opcionalmente el atributo pnodeUserId. El valor de este atributo es el ID de usuario que utiliza el nodo Connect:Direct que especifica el atributo name del elemento <tns:pnode> para enviar el proceso de Connect:Direct. Si no especifica el atributo pnodeUserId, el nodo Connect:Direct utiliza el ID de usuario que especifica el atributo cdUserId para someter el proceso Connect:Direct.

- Opcionalmente puede especificar el atributo pnodePassword. El valor de este atributo es la contraseña asociada con el ID de usuario que especifica el atributo pnodeUserId.

Si no hay ningún elemento de usuario que coincida con el ID de usuario MQMD, la transferencia falla.

3. Opcional: Puede incluir uno o más elementos <tns:snode> como elementos hijo del elemento <tns:user>. El elemento <tns:snode> especifica las credenciales que utiliza el nodo Connect:Direct que forma parte del puente Connect:Direct. Estas credenciales son el ID de usuario y contraseña que el nodo de puente Connect:Direct utiliza para conectarse al nodo Connect:Direct que es el origen o el destino de la transferencia de archivos.

   Inserte uno o varios de los elementos siguientes en el archivo:

```
<tns:snode name="name"
           pattern="pattern"
           userId="userId"
           password="password"/>
```

   donde:

   - *name* es un patrón a emparejar con el nombre del nodo Connect:Direct que es el origen o el destino de la transferencia de archivos.
   - *pattern* especifica si el patrón especificado en el atributo name es una expresión comodín o una expresión regular Java. El valor del atributo pattern puede ser wildcard o regex. Si no se especifica este atributo, el valor predeterminado es wildcard.
   - *userId* es el ID de usuario que utiliza el nodo Connect:Direct que especifica el atributo name del elemento <tns:pnode> para conectarse a un nodo Connect:Direct que coincide con el patrón especificado por el atributo name de <tns:snode>.
   - *password* es la contraseña asociada con el ID de usuario que especifica el atributo userId.

   Si ningún elemento <tns:snode> coincide con el nodo secundario de la transferencia de archivos, esto no hará que la transferencia falle. La transferencia se inicia y no se especifica ningún ID de usuario ni contraseña para utilizarlos con snode.

Resultados

Al buscar una coincidencia de patrón para nombres de usuario o nombres de nodos Connect:Direct, 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” en la página 844

Configure el puente Connect:Direct para transferir archivos entre una red de Managed File Transfer y una red de Connect:Direct. Los componentes del puente Connect:Direct son un nodo Connect:Direct y un agente de Managed File Transfer que está dedicado a comunicarse con dicho nodo. Este agente se denomina el agente de puente Connect:Direct.

Referencia relacionada

Formato del archivo de credenciales Connect:Direct

fteCreateCDAgent: crear un agente de puente Connect:Direct

Correlación de credenciales en Connect:Direct mediante clases de salida

Si no desea utilizar la función de correlación de credenciales predeterminada del agente de puente Connect:Direct, puede correlacionar las credenciales de usuario de Managed File Transfer con las credenciales de usuario en un nodo Connect:Direct escribiendo su propia salida de usuario. Si configura sus propias salidas de usuario de correlación de credenciales, se inhabilita la función de correlación de credenciales predeterminada.

Acerca de esta tarea

Las salidas de usuario que cree para correlacionar credenciales de Connect:Direct deben implementar la interfaz `com.ibm.wmqfte.exitroutine.api.ConnectDirectCredentialExit`. Si desea más información, consulte Interfaz CDCredentialExit.java.

Configuración de IBM MQ Console y REST API

El servidor mqweb que aloja IBM MQ Console y REST API se proporciona con una configuración predeterminada. Para utilizar cualquiera de estos componentes es necesario realizar una serie de tareas de configuración, como la configuración de seguridad para permitir a los usuarios iniciar la sesión. Este tema describe todas las opciones de configuración que están disponibles.

Procedimiento

Configuración básica para el servidor mqweb

Antes de poder empezar a utilizar la REST API o la IBM MQ Console, debe instalar los componentes correctos y configurar el servidor mqweb que aloja la REST API o la IBM MQ Console.

Acerca de esta tarea

El procedimiento para esta tarea se centra en una configuración básica para el servidor mqweb, de modo que pueda iniciarse rápidamente con la REST API y la IBM MQ Console. Los pasos para configurar la seguridad describen cómo configurar un registro de usuario básico, pero hay otras opciones para configurar usuarios y roles. Para obtener más información sobre cómo configurar la seguridad para el servidor mqweb, consulte Seguridad deIBM MQ Console y REST API.

Nota: Debe tener acceso al archivo `mqwebuser.xml` para completar este procedimiento:

- z/OS En z/OS, debe ser un usuario que tenga acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario privilegiado para acceder al archivo `mqwebuser.xml`.
- Linux V 9.4.0 Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Procedimiento

1. Instale el componente IBM MQ Console y REST API:
   - AIX En AIX, instale el conjunto de archivos `mqm.web.rte`. Para obtener más información sobre cómo instalar conjuntos de archivos en AIX, consulte Tareas de instalación de AIX.
   - IBM i En IBM i, instale el componente WEB. Para utilizar esta característica, también debe instalar 5724L26 IBM MQ Java Messaging and Web Services, y los requisitos previos de 5770JV1 Java SE 8. Para obtener más información sobre cómo instalar características en IBM i, consulte Tareas de instalación de IBM i.
   - Linux En Linux, instale el componente `MQSeriesWeb`. Para obtener más información sobre cómo instalar componentes en Linux, consulte Tareas de instalación de Linux.

     V 9.4.0 A partir de IBM MQ 9.4.0, también puede ejecutar el servidor mqweb en una instalación autónoma de IBM MQ Web Server en Linux. Para obtener más información sobre la instalación del IBM MQ Web Server, consulte Instalación del IBM MQ Web Serverautónomo.
   - Windows En Windows, instale la característica `Web Administration`. Para obtener más información sobre cómo instalar características en Windows, consulte Tareas de instalación de Windows.
   - z/OS Instale la característica `IBM MQ for z/OS UNIX System Services Web Components`. Para obtener más información sobre cómo instalar componentes y características en z/OS, consulte Tareas de instalación de z/OS.
2. Cree el servidor mqweb que aloja IBM MQ Console y REST API.
   - z/OS En z/OS, ejecute el script **crtmqweb** .

     El script crea el directorio de usuario de WebSphere Liberty que contiene los archivos de registro y configuración del servidor mqweb. Si desea más información sobre cómo ejecutar el script **crtmqweb**, consulte “Creación del servidor mqweb” en la página 963.
   - Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server , siga los pasos de “Configuración del IBM MQ Web Server autónomo” en la página 854.
   - En todos los demás entornos, no es necesario que realice ninguna acción para crear el servidor mqweb.
3. z/OS

   En z/OS, cree un procedimiento catalogado para iniciar el servidor mqweb.

   Para obtener más información, consulte “Creating a procedure for the mqweb server” en la página 966.
4. Sustituya el archivo de configuración existente, `mqwebuser.xml` por el archivo de ejemplo de registro básico que está configurado para ofrecer seguridad básica. Copie el archivo `basic_registry.xml` del directorio *MQ_INSTALLATION_PATH*`/web/mq/samp/configuration` en el directorio adecuado para el sistema y cambie el nombre del archivo por `mqwebuser.xml`:
   - En una instalación de IBM MQ , copie el archivo en el directorio siguiente:
     - Linux AIX En AIX and Linux: `/var/mqm/web/installations/`*`installationName`*`/servers/mqweb`
     - Windows En Windows: *MQ_DATA_PATH*`\web\installations\`*`installationName`*`\servers\mqweb`

       Donde *VÍA_DATOS_MQ* es la vía de acceso de datos de IBM MQ, esta vía de acceso es la vía de datos que se selecciona durante la instalación de IBM MQ. De forma predeterminada, esta vía de acceso es `C:\ProgramData\IBM\MQ`.

– z/OS En z/OS: *WLP_user_directory*/servers/mqweb

Donde *directorio_usuario_WLP* es el directorio que se ha especificado cuando el script **crtmqweb** se ejecutó para crear la definición de servidor mqweb.

- Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server : *MQ_OVERRIDE_DATA_PATH*/web/installations/MQWEBINST/servers/mqweb

  donde *MQ_OVERRIDE_DATA_PATH* es el directorio de datos de IBM MQ Web Server al que apunta la variable de entorno **MQ_OVERRIDE_DATA_PATH** .

El archivo de ejemplo basic_registry.xml configura cuatro usuarios:

mqadmin
Un usuario administrativo que es miembro del rol MQWebAdmin .

mqreader
Un usuario administrativo de sólo lectura que es miembro del rol MQWebAdminRO .

mftadmin
Un usuario administrativo que es miembro del rol MFTWebAdmin .

mftreader
Un usuario administrativo de sólo lectura que es miembro del rol MFTWebAdminRO .

Todos los usuarios también son miembros del rol MQWebUser .

Si desea más información sobre los roles disponibles, consulte Roles en la IBM MQ Console y la REST API

5. Opcional: Edite el archivo mqwebuser.xml para añadir más usuarios y grupos. Asigne a estos usuarios y grupos los roles apropiados para que estén autorizados para utilizar la REST API o la IBM MQ Console. También puede cambiar las contraseñas para los usuarios que están definidos de forma predeterminada, y codifique las nuevas contraseñas. Si desea más información, consulte Configuración de usuarios y roles.

   Nota:

   - z/OS En z/OS, si añade usuarios al rol MQWebUser , también debe otorgar al ID de usuario de la tarea iniciada mqweb acceso de usuario alternativo a los ID de usuario con el rol MQWebUser . Por ejemplo:

     ```
     RDEFINE MQADMIN hlq.ALTERNATE.USER.userId UACC(NONE)
     PERMIT hlq.ALTERNATE.USER.userId CLASS(MQADMIN) ACCESS(UPDATE) ID(mqwebUserId)
     ```

   - Multi z/OS Para completar los pasos para empezar con messaging REST API, debe añadir un usuario al archivo mqwebuser.xml. Este usuario debe tener el mismo nombre que un usuario de IBM MQ existente en el sistema. Siguiendo el mismo formato que los demás usuarios del archivo xml, añada el ID de usuario y una contraseña después de la siguiente línea en el archivo xml: <user name="mftreader" password="mftreader"/>.

6. Establezca el entorno para que apunte a la configuración del servidor mqweb.

   - z/OS En z/OS, establezca la variable de entorno WLP_USER_DIR para que la variable señale 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 al mandato **crtmqweb** . Por ejemplo:

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

     Para obtener más información, consulte “Creación del servidor mqweb” en la página 963.

- Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server , establezca la variable de entorno **MQ_OVERRIDE_DATA_PATH** en el directorio de datos IBM MQ Web Server .

  Por ejemplo, si elige utilizar `/var/mqweb` como directorio de datos de IBM MQ Web Server , emita el mandato siguiente:

  ```
  export MQ_OVERRIDE_DATA_PATH=/var/mqweb
  ```

- En todos los demás entornos, no es necesario que realice ninguna acción para establecer el entorno.

7. De forma predeterminada, la REST API y la IBM MQ Console solo están disponibles desde el mismo host que el servidor mqweb. Habilite conexiones remotas con el servidor mqweb especificando el mandato siguiente:

   ```
   setmqweb properties -k httpHost -v hostname
   ```

   Donde *nombrehost* especifica la dirección IP, el nombre de host del servidor de nombres de dominio (DNS) con el sufijo del nombre de dominio, o el nombre de host del DNS del servidor donde está instalado IBM MQ. Utilice un asterisco, *, entre comillas dobles, para especificar todas las interfaces de red disponibles, tal como se indica en el ejemplo siguiente:

   ```
   setmqweb properties -k httpHost -v "*"
   ```

8. Opcional: De forma predeterminada, la administrative REST API para MFT no está habilitada. Si desea utilizar esta característica, debe habilitarla y configurar un gestor de colas de coordinación.

   a) Habilite la administrative REST API para MFT especificando el mandato siguiente:

   ```
   setmqweb properties -k mqRestMftEnabled -v true
   ```

   b) Configure qué gestor de colas es el gestor de colas de coordinación especificando el mandato siguiente:

   ```
   setmqweb properties -k mqRestMftCoordinationQmgr -v qmgrName
   ```

   Donde *NombreGestorColas* es el nombre del gestor de colas de coordinación.

   c) Para habilitar las llamadas POST, configure qué gestor de colas es el gestor de colas de mandatos especificando el mandato siguiente:

   ```
   setmqweb properties -k mqRestMftCommandQmgr -v qmgrName
   ```

   donde *NombreGestorColas* es el nombre del gestor de colas de mandatos.

9. Inicie el servidor mqweb que da soporte a la REST API y la IBM MQ Console:

- ALW En AIX, Linux, and Windows, como usuario privilegiado, especifique el mandato siguiente:

  ```
  strmqweb
  ```

- IBM i En IBM i, como usuario privilegiado, especifique el mandato siguiente en Qshell:

  ```
  /QIBM/ProdData/mqm/bin/strmqweb
  ```

- z/OS En z/OS, inicie el procedimiento que ha creado en “Creating a procedure for the mqweb server” en la página 966.

  Los mensajes siguientes se emiten a STDOUT DD para indicar que el servidor mqweb se ha iniciado satisfactoriamente.

  ```
  [AUDIT   ] MQWB2019I: MQ Console level: 9.2.4 - V924-CD924-L211028
  [AUDIT   ] MQWB0023I: MQ REST API level: 9.2.4 - V924-CD924-L211028
  [AUDIT   ] CWWKZ0001I: Application com.ibm.mq.rest started in 1.763 seconds.
  ```

```
[AUDIT   ] CWWKZ0001I: Application com.ibm.mq.console started in 2.615 seconds.
[AUDIT   ] CWWKF0011I: The mqweb server is ready to run a smarter planet. The mqweb
server started in 10.016 seconds.
```

Puede detener el servidor mqweb en cualquier momento deteniendo la tarea iniciada del servidor mqweb en z/OS o utilizando el mandato **endmqweb**. No obstante, si el servidor mqweb no se está ejecutando, no puede utilizar REST API o IBM MQ Console.

10. z/OS

Opcional: En z/OS, si desea permitir que los productos de automatización del sistema atrapen los mensajes MQWB2019I y MQWB0023I que se emiten cuando se inician IBM MQ Console y REST API, configure el servidor mqweb para escribir estos mensajes en la consola de MVS. Para configurar el servidor mqweb para escribir los mensajes MQWB2019I y MQWB0023I en la consola de MVS, edite el archivo `mqwebuser.xml` que ha creado en el paso "4" en la página 851 y añada la siguiente línea al archivo:

```
<zosLogging enableLogToMVS="true" wtoMessage="MQWB2019I,MQWB0023I"/>
```

Para obtener más información sobre la configuración del Registro de z/OS en el servidor mqweb, consulte Registro de z/OS (zosLogging).

Qué hacer a continuación

1. Configure los valores del servidor mqweb, incluyendo habilitar conexiones HTTP y cambiar el número de puerto. Para obtener más información, consulte "Configuración de IBM MQ Console y REST API" en la página 850.
2. Opcionalmente, configure la REST API:
   a. Configure Cross Origin Resource Sharing (CORS) para REST API. De forma predeterminada, no puede acceder a REST API desde recursos web que no estén alojados en el mismo dominio que REST API. Es decir, las solicitudes entre orígenes no están habilitadas. Puede configurar Cross Origin Resource Sharing (CORS) para permitir las solicitudes entre orígenes a partir de los URL especificados. Para obtener más información, consulte Configuración de CORS para REST API.
   b. Configure la REST API para MFT. Para obtener más información, consulte "Configuración de la REST API para MFT" en la página 875.
3. Utilice la REST API o la IBM MQ Console:
   - Iniciación a la administrative REST API
   - Iniciación a la messaging REST API
   - Iniciación a la IBM MQ Console

Linux V 9.4.0 Configuración del IBM MQ Web Server autónomo

A partir de IBM MQ 9.4.0, puede ejecutar el servidor mqweb que aloja IBM MQ Console y REST API en una instalación autónoma de IBM MQ Web Server .

Antes de empezar

El IBM MQ Web Server autónomo sólo está disponible en Linux.

Para poder configurar el servidor mqweb, debe instalar el IBM MQ Web Server siguiendo los pasos de Instalación del IBM MQ Web Serverautónomo.

Acerca de esta tarea

Siga el procedimiento de esta tarea para crear y configurar un nuevo servidor mqweb que se ejecute en una instalación autónoma de IBM MQ Web Server . Puede configurar más de un servidor mqweb para que se ejecute en una instalación autónoma de IBM MQ Web Server repitiendo este procedimiento.

Procedimiento

1. Cree el directorio de datos IBM MQ Web Server .

   El directorio de datos se utiliza para almacenar los archivos de configuración y de registro para el servidor mqweb que ejecuta IBM MQ Console y REST API. Puede utilizar cualquier directorio que elija como directorio de datos de IBM MQ Web Server .

   Al ID de usuario que utilice para iniciar el servidor mqweb se le debe otorgar acceso de lectura y escritura al directorio de datos.

2. Establezca la variable de entorno **MQ_OVERRIDE_DATA_PATH** en el directorio de datos que ha creado en el paso "1" en la página 855.
   Por ejemplo, si elige utilizar `/var/mqweb` como directorio de datos de IBM MQ Web Server , emita el mandato siguiente:

   ```
   export MQ_OVERRIDE_DATA_PATH=/var/mqweb
   ```

3. Utilice el mandato **setmqenv** para configurar el entorno de IBM MQ .

   Vaya al directorio `bin` del directorio de instalación de IBM MQ Web Server y, a continuación, emita el mandato siguiente:

   ```
   . setmqenv -s
   ```

4. Utilice el mandato **crtmqdir** para crear los directorios y archivos IBM MQ , en el directorio de datos. Los archivos que se crean incluyen una definición de plantilla para el servidor mqweb.

   Emita el mandato siguiente:

   ```
   crtmqdir -s -f
   ```

5. Opcional: Si este servidor mqweb es el primero que ha creado para ejecutar con esta instalación del IBM MQ Web Serverautónomo, utilice el mandato **mqlicense** para revisar y aceptar la licencia de IBM MQ .

   Debe ejecutar este mandato como un usuario que tenga acceso de escritura al directorio de instalación de IBM MQ Web Server .

   Por ejemplo, emita el mandato siguiente para ver la licencia de IBM MQ :

   ```
   mqlicense
   ```

   Para obtener más información, consulte mqlicense.

6. Opcional: Para migrar un servidor mqweb existente para que se ejecute en la instalación de IBM MQ Web Server autónoma recién configurada, realice los pasos siguientes:

   a. Realice una copia de seguridad de la configuración del servidor mqweb existente.

   b. Restaure los archivos en el directorio *MQ_OVERRIDE_DATA_PATH*`/web/installations/MQWEBINST` , donde *MQ_OVERRIDE_DATA_PATH* es el directorio de datos de IBM MQ Web Server que ha creado en el paso "1" en la página 855.

   Para obtener más información, consulte "Copia de seguridad y restauración de la configuración del servidor mqweb" en la página 885.

   Nota: Algunas características de IBM MQ Console y REST API no están disponibles en una instalación autónoma de IBM MQ Web Server . Si migra un servidor mqweb desde una instalación de IBM MQ a una instalación autónoma de IBM MQ Web Server , estas características no se pueden utilizar después de la migración. Para obtener más información sobre las restricciones que se aplican en una instalación autónoma de IBM MQ Web Server , consulte IBM MQ Console y REST API.

Qué hacer a continuación

Configure el servidor mqweb siguiendo los pasos que se describen en "Configuración básica para el servidor mqweb" en la página 850.

Configuración de la seguridad

Puede configurar la seguridad para IBM MQ Console y REST API editando el archivo `mqwebuser.xml`. Puede configurar y autenticar usuarios configurando un registro de usuario básico, o un registro LDAP, o cualquiera de los demás tipos de registro que se proporcionan con WebSphere Liberty. A continuación, puede autorizar a esos usuarios asignando un rol a los usuarios y grupos.

Acerca de esta tarea

Para configurar la seguridad para IBM MQ Console y REST API, debe configurar los usuarios y grupos. A continuación, estos usuarios y grupos se pueden autorizar para utilizar IBM MQ Console o REST API, o ambos. Para obtener más información sobre la configuración de usuarios y grupos y la autenticación y autorización de usuarios, consulte Seguridad deIBM MQ Console y REST API.

Cuando los usuarios se autentican con IBM MQ Console, se genera una señal LTPA. Esta señal permite al usuario utilizar IBM MQ Console sin volver a autenticarse hasta que caduque la señal.

Si utiliza la autenticación basada en señal con la REST API, se genera una señal LTPA cuando el usuario inicia sesión utilizando el recurso `/login` de REST API con el método HTTP POST. Puede configurar cuándo caduca esta señal, y si se puede utilizar esta señal para ambas conexiones, HTTP y HTTPS. Para obtener más información, consulte “Configuración de la señal LTPA” en la página 864.

Procedimiento

- Seguridad deIBM MQ Console y REST API
- “Configuración de la señal LTPA” en la página 864

Configuración del nombre de host HTTP

De forma predeterminada, el servidor mqweb que aloja IBM MQ Console y REST API está configurado para permitir sólo conexiones locales. Es decir, sólo se puede acceder a IBM MQ Console y REST API en el sistema en el que están instalados IBM MQ Console y REST API. Puede configurar el nombre de host para permitir las conexiones remotas utilizando el mandato **setmqweb**.

Antes de empezar

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.
- V 9.4.0 Linux Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Atención: V 9.4.0 Linux

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en una instalación autónoma de IBM MQ Web Server , debe establecer la variable de entorno **MQ_OVERRIDE_DATA_PATH** en el directorio de datos IBM MQ Web Server .

Procedimiento

- Consulte la configuración actual del nombre de host HTTP utilizando el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

  El campo `httpHost` muestra el nombre de host HTTP.
- Establezca el nombre de host HTTP utilizando el mandato siguiente:

  ```
  setmqweb properties -k httpHost -v hostName
  ```

  Donde *nombrehost* especifica la dirección IP, el nombre de host del servidor de nombres de dominio (DNS) con el sufijo del nombre de dominio, o el nombre de host del DNS del servidor donde está instalado IBM MQ. Utilice un asterisco entre dobles comillas para especificar todas las interfaces de red disponibles. Utilice el valor `localhost` solo para permitir conexiones locales.
- Desconfigure el nombre de host HTTP utilizando el mandato siguiente:

  ```
  setmqweb properties -k httpHost -d
  ```

Configuración de los puertos HTTP y HTTPS

De forma predeterminada, el servidor mqweb que aloja IBM MQ Console y REST API utiliza el puerto HTTPS 9443. El puerto que está asociado con conexiones HTTP está inhabilitado. Puede habilitar el puerto HTTP, configurar un puerto HTTPS distinto o inhabilitar el puerto HTTP o HTTPS. Puede configurar los puertos utilizando el mandato **setmqweb**.

Antes de empezar

Si habilita el puerto HTTP y está utilizando la autenticación basada en señales, debe habilitar la misma señal LTPA que se va a utilizar para ambas conexiones, HTTP y HTTPS. Para obtener más información, consulte “Configuración de la señal LTPA” en la página 864.

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.
- V 9.4.0 Linux Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Atención: V 9.4.0 Linux

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en una instalación autónoma de IBM MQ Web Server , debe establecer la variable de entorno **MQ_OVERRIDE_DATA_PATH** en el directorio de datos IBM MQ Web Server .

Atención: De forma predeterminada, el servidor mqweb requiere que las señales LTPA estén protegidas para todas las solicitudes. Si el servidor mqweb está configurado para requerir que las señales LTPA estén protegidas, no puede completar las acciones siguientes cuando se conecte al puerto HTTP:

- Inicie la sesión en IBM MQ Console.
- Utilice la autenticación basada en señal con REST API.

Para permitir que las solicitudes HTTP utilicen señales LTPA, establezca el valor de la propiedad **secureLTPA** en `false`. Para obtener más información, consulte "Configuración de la señal LTPA" en la página 864.

Procedimiento

- Consulte la configuración actual de los puertos HTTP y HTTPS utilizando el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

  El campo `httpPort` muestra el puerto HTTP y el campo `httpsPort` muestra el puerto HTTPS.
- Habilite o configure el puerto HTTP: utilizando el mandato siguiente:
  - Habilite o establezca el puerto HTTP utilizando el mandato siguiente:

    ```
    setmqweb properties -k httpPort -v portNumber
    ```

    donde *númeroPuerto* especifica el puerto que desea utilizar para las conexiones HTTP. Puede inhabilitar el puerto utilizando un valor de `-1`.
  - Restablezca el valor de puerto HTTP en el valor predeterminado de `-1` utilizando el mandato siguiente:

    ```
    setmqweb properties -k httpPort -d
    ```
- Configure el puerto HTTPS:
  - Establezca el número de puerto HTTPS utilizando el mandato siguiente:

    ```
    setmqweb properties -k httpsPort -v portNumber
    ```

    donde *númeroPuerto* especifica el puerto que desea utilizar para las conexiones HTTPS. Puede inhabilitar el puerto utilizando un valor de `-1`.
  - Restablezca el número de puerto HTTPS en el valor predeterminado de 9443 utilizando el mandato siguiente:

    ```
    setmqweb properties -k httpsPort -d
    ```

Configuración del tiempo de espera de respuesta

De forma predeterminada, IBM MQ Console y REST API exceden el tiempo de espera si el tiempo necesario para devolver una respuesta a un cliente es más de 30 segundos. Puede configurar la IBM MQ Console y la REST API para utilizar un valor de tiempo de espera diferente utilizando el mandato **setmqweb**.

Antes de empezar

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.
- V 9.4.0 Linux Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Atención: V 9.4.0 Linux

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en una instalación autónoma de IBM MQ Web Server , debe establecer la variable de entorno **MQ_OVERRIDE_DATA_PATH** en el directorio de datos IBM MQ Web Server .

Procedimiento

- Consulte la configuración actual del tiempo de espera de solicitud utilizando el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

  El campo `mqRestRequestTimeout` muestra el valor actual del tiempo de espera de respuesta. Para obtener más información, consulte Propiedades de dspmqweb.
- Establezca el tiempo de espera de solicitud utilizando el mandato siguiente:

  ```
  setmqweb properties -k mqRestRequestTimeout -v timeout
  ```

  Donde *tiempo_espera* especifica, en segundos, el tiempo de espera antes de que éste se supere.
- Restablezca el tiempo de espera de solicitud en el valor predeterminado de 30 segundos utilizando el mandato siguiente:

  ```
  setmqweb properties -k mqRestRequestTimeout -d
  ```

Configuración del inicio automático

De forma predeterminada, IBM MQ Console se inicia automáticamente cuando se inicia el servidor mqweb. Puede configurar si la IBM MQ Console y la REST API se van a iniciar automáticamente utilizando el mandato **setmqweb**.

Antes de empezar

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

- V 9.4.0 Linux Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Atención: V 9.4.0 Linux

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en una instalación autónoma de IBM MQ Web Server , debe establecer la variable de entorno **MQ_OVERRIDE_DATA_PATH** en el directorio de datos IBM MQ Web Server .

Procedimiento

- Consulte la configuración actual del inicio automático mediante el mandato siguiente:

  `dspmqweb properties -a`

  El campo `mqRestAutostart` muestra si REST API se inicia automáticamente, y el campo `mqConsoleAutostart` muestras si IBM MQ Console se inicia automáticamente.
- Configure si la IBM MQ Console se inicia automáticamente mediante el mandato siguiente:

  `setmqweb properties -k mqConsoleAutostart -v start`

  donde *start* es el valor `true` si desea que la IBM MQ Console se inicie automáticamente o `false`, en caso contrario.
- Configure si la REST API se inicia automáticamente mediante el mandato siguiente:

  `setmqweb properties -k mqRestAutostart -v start`

  donde *start* es el valor `true` si desea que la REST API se inicie automáticamente o `false`, en caso contrario.

Configuración del registro

Puede configurar los niveles de registro, el tamaño máximo del archivo de registro y el número máximo de archivos de registro utilizados por el servidor mqweb que aloja IBM MQ Console y REST API. Puede configurar el registro utilizando el mandato **setmqweb**.

Antes de empezar

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

- V 9.4.0 Linux Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Atención: V 9.4.0 Linux

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en una instalación autónoma de IBM MQ Web Server , debe establecer la variable de entorno **MQ_OVERRIDE_DATA_PATH** en el directorio de datos IBM MQ Web Server .

Acerca de esta tarea

El servidor mqweb graba mensajes de registro y rastreo en los siguientes archivos de registro:

console.log y messages.log
Estos archivos contienen mensajes emitidos por el IBM MQ Console, el REST APIy el servidor mqweb que ejecuta estos componentes.

trace.log
Este archivo contiene el rastreo para IBM MQ Console y REST API. El rastreo se graba en este archivo sólo si el rastreo está habilitado.

Los archivos de registro para el servidor mqweb se pueden encontrar en uno de los directorios siguientes:

- En una instalación de IBM MQ :
  - Linux AIX En AIX o Linux: `/var/mqm/web/installations/`*`installationName`*`/servers/mqweb/logs`
  - Windows En Windows: *MQ_DATA_PATH*`\web\installations\`*`installationName`*`\servers\mqweb\logs`, donde *MQ_DATA_PATH* es la vía de acceso de datos de IBM MQ . Esta vía de acceso es la vía de acceso de datos que se selecciona durante la instalación de IBM MQ. De forma predeterminada, esta vía de acceso es `C:\ProgramData\IBM\MQ`.
  - z/OS En z/OS: *WLP_user_directory*`/servers/mqweb/logs`

    donde *directorio_usuario_WLP* es el directorio que se ha especificado cuando se ejecutó el script **crtmqweb** para crear la definición del servidor mqweb.
- Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server : *MQ_OVERRIDE_DATA_PATH*`/web/installations/MQWEBINST/servers/mqweb/logs`

  donde *MQ_OVERRIDE_DATA_PATH* es el directorio de datos de IBM MQ Web Server al que apunta la variable de entorno **MQ_OVERRIDE_DATA_PATH** .

Los archivos de rastreo de mensajería para el código REST API de mensajería que se ejecuta en el servidor mqweb se pueden encontrar en uno de los directorios siguientes:

- En una instalación de IBM MQ :

  - Linux AIX En AIX o Linux: `/var/mqm/web/installations/installationName/servers/mqweb`
  - Windows En Windows: *MQ_DATA_PATH*`\web\installations\`*installationName*`\servers\mqweb`, donde *MQ_DATA_PATH* es la vía de acceso de datos de IBM MQ . Esta vía de acceso es la vía de acceso de datos que se selecciona durante la instalación de IBM MQ. De forma predeterminada, esta vía de acceso es `C:\ProgramData\IBM\MQ`.
  - z/OS En z/OS: *WLP_user_directory*`/servers/mqweb`

    donde *directorio_usuario_WLP* es el directorio que se ha especificado cuando se ejecutó el script **crtmqweb** para crear la definición del servidor mqweb.
- Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server : *MQ_OVERRIDE_DATA_PATH*`/web/installations/MQWEBINST/servers/mqweb`

  donde *MQ_OVERRIDE_DATA_PATH* es el directorio de datos de IBM MQ Web Server al que apunta la variable de entorno **MQ_OVERRIDE_DATA_PATH** .

Para obtener más información sobre cómo habilitar el rastreo para:

- REST API, consulte Rastreo de REST API
- IBM MQ Console, consulte Rastreo de IBM MQ Console

Procedimiento

- Consulte la configuración actual del registro de REST API utilizando el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

  - El campo `maxTraceFileSize` muestra el tamaño máximo del archivo de registro
  - El campo `maxTraceFiles` muestra el número máximo de archivos de registro
  - El campo `traceSpec` muestra el nivel de rastreo que se utiliza
  - El campo `maxMsgTraceFileSize` muestra el tamaño máximo del archivo de rastreo de mensajería
  - El campo `maxMsgTraceFiles` muestra el número máximo de archivos de rastreo de mensajería
- Configure el tamaño máximo de los archivos `messages.log` y `trace.log` :
  - Establezca el tamaño máximo del archivo de registro utilizando el mandato siguiente:

    ```
    setmqweb properties -k maxTraceFileSize -v size
    ```

    donde *tamaño* especifica el tamaño, en MB, quede puede alcanzar cada archivo de registro.
  - Restablezca el tamaño máximo de archivo de registro al valor predeterminado de 20 MB utilizando el mandato siguiente:

    ```
    setmqweb properties -k maxTraceFileSize -d
    ```
- Configure el número máximo de los archivos `messages.log` y `trace.log` :
  - Establezca el número máximo de cada archivo de registro utilizando el mandato siguiente:

    ```
    setmqweb properties -k maxTraceFiles -v max
    ```

    donde *máx* especifica el número máximo de archivos.
  - Restablezca el número máximo de cada archivo de registro al valor predeterminado de 2 utilizando el mandato siguiente:

    ```
    setmqweb properties -k maxTraceFiles -d
    ```
- Configure el tamaño máximo del archivo de rastreo de mensajería:
  - Establezca el tamaño máximo del archivo de rastreo de mensajería utilizando el mandato siguiente:

```
setmqweb properties -k maxMsgTraceFileSize -v size
```

donde *tamaño* especifica el tamaño, en MB, que puede alcanzar cada archivo de rastreo de mensajería.

- Restablezca el tamaño máximo del archivo de rastreo de mensajería al valor predeterminado de 200 MB utilizando el mandato siguiente:

```
setmqweb properties -k maxMsgTraceFileSize -d
```

- Configure el número máximo de archivos de rastreo de mensajería para utilizar:
  - Establezca el número máximo de archivos para utilizar el rastreo de mensajería utilizando el mandato siguiente:

```
setmqweb properties -k maxMsgTraceFiles -v max
```

donde *máx* especifica el número máximo de archivos.

  - Restablezca el número máximo de archivos para utilizar para el rastreo de mensajería en el valor predeterminado de 5 utilizando del mandato siguiente:

```
setmqweb properties -k maxMsgTraceFiles -d
```

- Configure el nivel de rastreo que escribe el servidor mqweb:
  - Establezca la especificación de rastreo que se utiliza utilizando el mandato siguiente:

```
setmqweb properties -k traceSpec -v level
```

donde *nivel* es uno de los valores que se listan en Tabla 52 en la página 863. La tabla describe los niveles de registro, ordenados por el aumento del nivel de detalle. Cuando se habilita un nivel de registro, también se habilita cada nivel antes del mismo. Por ejemplo, si habilita el nivel de registro ***=warning**, también habilita los niveles de registro ***=severe** y ***=fatal**.

Cambie este valor cuando se lo solicite el servicio de soporte de IBM .

  - Restablezca la especificación de rastreo que se utiliza en el valor predeterminado de ***=info** utilizando el mandato siguiente:

```
setmqweb properties -k traceSpec -d
```

Tabla 52. Niveles de registro válidos

Valor	**Nivel de registro aplicado**
*=off	El registro esta desactivado.
*=fatal	La tarea no puede continuar y el componente, la aplicación y el servidor no pueden funcionar.
*=severe	La tarea no puede continuar pero el componente, la aplicación y el servidor pueden seguir funcionando. Este nivel también puede indicar un error irrecuperable inminente.
*=warning	Error potencial o error inminente. Este nivel también puede indicar una anomalía progresiva (por ejemplo, la pérdida potencial de recursos).
*=audit	Suceso significativo que afecta al estado del servidor o a los recursos
*=info	Información general que describe el progreso de tarea en conjunto
*=config	Estado o cambio de configuración
*=detail	Información general que describe en detalle el progreso de subtarea

Tabla 52. Niveles de registro válidos (continuación)	
Valor	**Nivel de registro aplicado**
*=fine	Información de rastreo: rastreo general + entrada de método, salida y valores de retorno
*=finer	Información de rastreo: rastreo detallado
*=finest	Información de rastreo: rastreo más detallado que incluye todos los detalles necesarios para depurar problemas
* =todos	Se registran todos los sucesos

Configuración de la señal LTPA

Las señales LTPA se pueden utilizar para evitar que un usuario proporcione las credenciales de nombre de usuario y contraseña en cada solicitud al servidor mqweb. Puede configurar el nombre de la cookie de la señal LTPA, el intervalo de caducidad para las señales de autenticación LTPA y configurar si las señales LTPA pueden ser utilizadas por conexiones HTTP, utilizando el mandato **setmqweb**.

Antes de empezar

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.
- V 9.4.0 Linux Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Nota: Si utiliza IBM MQ Console y la autenticación de señal con REST API, el intervalo de caducidad se comparte.

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Atención: V 9.4.0 Linux

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en una instalación autónoma de IBM MQ Web Server , debe establecer la variable de entorno **MQ_OVERRIDE_DATA_PATH** en el directorio de datos IBM MQ Web Server .

Acerca de esta tarea

Cuando los usuarios inician una sesión en IBM MQ Console, se genera una señal LTPA. Si utiliza la autenticación basada en señal con la REST API, se genera una señal LTPA cuando el usuario inicia sesión

utilizando el recurso `/login` de REST API con el método HTTP POST. Esta señal se devuelve en una cookie. La señal se utiliza para autenticar el usuario sin que deba iniciar una sesión de nuevo con su ID de usuario y contraseña, hasta que caduque la señal. El intervalo de caducidad predeterminado es de 120 minutos.

El nombre de la cookie que incluye la señal LTPA varía según la plataforma:

- MQ Appliance En IBM MQ Appliance, la señal LTPA es `LtpaToken2`. Este valor no se puede modificar.
- z/OS ALW De forma predeterminada, en todas las demás plataformas, el nombre de la cookie que incluye la señal LTPA empieza por `LtpaToken2` e incluye un sufijo que puede cambiar cuando se reinicia el servidor mqweb. Este nombre de cookie aleatorizado permite que se pueda ejecutar más de un servidor mqweb en el mismo sistema. Sin embargo, si desea que el nombre de la cookie siga siendo un valor coherente, puede especificar el nombre que tiene la cookie utilizando el mandato **setmqweb**.

z/OS IBM i ALW Si habilita ambos puertos, HTTP y HTTPS, se puede reutilizar una señal LTPA que se ha emitido para una solicitud HTTPS para una solicitud HTTP. Este comportamiento está inhabilitado de forma predeterminada, pero puede habilitar este comportamiento utilizando el mandato **setmqweb**.

Procedimiento

- Consulte la caducidad actual de la señal LTPA, el nombre de la cookie de la señal LTPA y si la señal LTPA se puede utilizar para solicitudes HTTP utilizando el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

  - El campo `ltpaCookieName` muestra el nombre de la cookie de la señal LTPA. Si no ha establecido un nombre de cookie, el valor de esta propiedad es `LtpaToken2_${env.MQWEB_LTPA_SUFFIX}` en AIX, Linux, and Windows , o `LtpaToken2_${httpsPort}` en z/OS, . La variable después del prefijo `LtpaToken2_` es utilizada por el servidor mqweb para generar un nombre exclusivo para la cookie. No puede establecer esta variable, pero puede cambiar el `ltpaCookieName` por un valor de su elección.
  - El campo `ltpaExpiration` muestra la hora de caducidad de la señal LTPA.
  - El campo `secureLtpa` se establece en `false` si las solicitudes HTTP pueden utilizar señales LTPA.
- Configure la caducidad de la señal LTPA:
  - Establezca la caducidad de la señal LTPA especificando el mandato siguiente:

    ```
    setmqweb properties -k ltpaExpiration -v time
    ```

    donde *tiempo* especifica el tiempo, en minutos, antes de la señal LTPA caduque y el usuario haya finalizado la sesión.
  - Restablezca la caducidad de la señal LTPA en el valor predeterminado de 120 minutos especificando el mandato siguiente:

    ```
    setmqweb properties -k ltpaExpiration -d
    ```
- z/OS ALW

  Configure el nombre de la cookie de la señal LTPA:
  - Establezca el nombre de la cookie de la señal LTPA especificando el mandato siguiente:

    ```
    setmqweb properties -k ltpaCookieName -v name
    ```

    donde *nombre* especifica un nombre exclusivo para la cookie de la señal LTPA.
  - Restablezca el nombre de la cookie de señal LTPA en el valor predeterminado, donde un prefijo de `LtpaToken2_` va seguido de caracteres aleatorios, especificando el mandato siguiente:

    ```
    setmqweb properties -k ltpaCookieName -d
    ```

- z/OS ALW
  Configure si la señal LTPA puede ser utilizada por conexiones HTTP especificando el mandato siguiente:

  ```
  setmqweb properties -k secureLtpa -v secure
  ```

  donde *secure* especifica si la señal LTPA se puede utilizar por ambas conexiones, conexiones HTTP no seguras y conexiones HTTPS seguras. Un valor de `false` permite que ambas conexiones, HTTP y HTTPS, utilicen la misma señal LTPA.

Configuración del comportamiento de conexión del gestor de colas remoto para IBM MQ Console

Al utilizar IBM MQ Console, puede crear conexiones con gestores de colas remotos. Es decir, puede conectarse a gestores de colas que no forman parte de la misma instalación que el servidor mqweb que ejecuta la IBM MQ Console. Hay una serie de opciones de configuración que puede establecer para controlar el comportamiento de las conexiones de gestores de colas remotos.

Antes de empezar

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.
- V 9.4.0 Linux Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Atención: V 9.4.0 Linux

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en una instalación autónoma de IBM MQ Web Server , debe establecer la variable de entorno **MQ_OVERRIDE_DATA_PATH** en el directorio de datos IBM MQ Web Server .

Acerca de esta tarea

Puede establecer las siguientes opciones de configuración:

- Indica si las conexiones de gestores de colas remotos están permitidas.
- Indica si las conexiones se pueden añadir utilizando la IBM MQ Console o sólo mediante la línea de mandatos.

- Indica si los gestores de colas locales se muestran en la IBM MQ Console cuando las conexiones de gestores de colas remotos están permitidas.
- Indica si las conexiones de gestores de colas remotos se establecen automáticamente cuando se inicia la IBM MQ Console o cuando hay un error de conexión.
- El período de tiempo entre cada renovación de la lista de gestores de colas remotos que se muestra en la IBM MQ Console.

Procedimiento

- Para ver los valores de configuración de conexión del gestor de colas remoto actual, especifique el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

  - El campo `mqConsoleRemoteSupportEnabled` indica si se permiten conexiones de gestor de colas remotas.
  - El campo `mqConsoleRemoteUIAdmin` indica si se pueden añadir conexiones de gestor de colas remotas utilizando IBM MQ Console.
  - El campo `mqConsoleRemoteAllowLocal` indica si se visualizan los gestores de colas locales.
  - El campo `mqConsoleRemotePollTime` indica cuántos segundos transcurren entre cada renovación de la lista de gestores de colas remotos.
- Para impedir o permitir conexiones de gestor de colas remotos con la IBM MQ Console, especifique el mandato siguiente:

  ```
  setmqweb properties -k mqConsoleRemoteSupportEnabled -v true or false
  ```

  donde *true* permite conexiones de gestor de colas remoto, o `false` impide conexiones de gestor de colas remoto.

  Nota: Linux V 9.4.0 Si el servidor mqweb se ejecuta en una instalación autónoma de IBM MQ Web Server , la propiedad **mqConsoleRemoteSupportEnabled** no es válida. El IBM MQ Web Server autónomo sólo da soporte a conexiones con gestores de colas remotos.
- Para impedir o permitir que se añadan conexiones de gestor de colas remoto mediante la IBM MQ Console o solo mediante la línea de mandatos, especifique el mandato siguiente:

  ```
  setmqweb properties -k mqConsoleRemoteUIAdmin -v true or false
  ```

  donde `true` permite añadir conexiones de gestor de colas remoto utilizando IBM MQ Console y la línea de mandatos, o `false` permite añadir conexiones de gestor de colas remoto sólo utilizando el mandato **setmqweb remote** en la línea de mandatos.
- Para impedir o permitir la visualización de gestores de colas locales en la IBM MQ Console cuando están permitidas las conexiones de gestores de colas remotos, especifique el mandato siguiente:

  ```
  setmqweb properties -k mqConsoleRemoteAllowLocal -v true or false
  ```

  donde `true` permite que se visualicen los gestores de colas locales o `false` oculta los gestores de colas locales.

  Nota: Linux V 9.4.0 Si el servidor mqweb se ejecuta en una instalación autónoma de IBM MQ Web Server , la propiedad **mqConsoleRemoteAllowLocal** no es válida. El IBM MQ Web Server autónomo sólo da soporte a conexiones con gestores de colas remotos.
- Para establecer el período de tiempo entre cada actualización de la lista de gestores de colas remotos que se muestra en la IBM MQ Console, especifique el mandato siguiente:

  ```
  setmqweb properties -k mqConsoleRemotePollTime -v seconds
  ```

  donde *segundos* se establece en un valor entero del número de segundos entre cada actualización de la lista de gestores de colas remotos.

Referencia relacionada

z/OS Pid de setmqweb

dspmqweb

Configuración de la pasarela de administrative REST API

Cuando la pasarela de administrative REST API está habilitada, puede realizar la administración remota con REST API utilizando un gestor de colas de pasarela. Puede configurar el gestor de colas que se utiliza como el gestor de colas de pasarela predeterminado, o puede impedir la administración remota inhabilitando la pasarela administrative REST API, utilizando el mandato **setmqweb**.

Antes de empezar

Nota: Linux V 9.4.0 Si el servidor mqweb se ejecuta en una instalación autónoma de IBM MQ Web Server , esta tarea no es aplicable. administrative REST API no está disponible en una instalación autónoma de IBM MQ Web Server .

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Acerca de esta tarea

Cuando el servidor mqweb se ejecuta en una instalación de IBM MQ , la pasarela administrative REST API está habilitada de forma predeterminada.

El gestor de colas de pasarela predeterminado 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.

Para obtener más información sobre la administración remota con REST API, consulte Administración remota utilizando REST API.

Procedimiento

- Consulte la configuración actual de la pasarela administrative REST API utilizando el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

El campo `mqRestGatewayEnabled` muestra si la pasarela está habilitada y el campo `mqRestGatewayQmgr` muestra el nombre del gestor de colas de pasarela predeterminado.

- Configure si la pasarela de administrative REST API está habilitada mediante el mandato siguiente:

  ```
  setmqweb properties -k mqRestGatewayEnabled -v enabled
  ```

  donde *enabled* es el valor **true** para habilitar la pasarela de administrative REST API, o **false** en caso contrario.
- Configure qué gestor de colas se utiliza como gestor de colas de pasarela predeterminado:
  - Establezca el gestor de colas de pasarela predeterminado mediante el mandato siguiente:

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

    donde *qmgrName* es el nombre de un gestor de colas en la misma instalación que el servidor mqweb.
  - Desconfigure el gestor de colas de pasarela predeterminado mediante el mandato siguiente:

    ```
    setmqweb properties -k mqRestGatewayQmgr -d
    ```

Configuración del messaging REST API

Puede configurar el messaging REST API de varias maneras. Puede elegir habilitar o inhabilitar la característica messaging REST API . Puede elegir el número máximo de conexiones agrupadas que puede utilizar el messaging REST API, y el comportamiento del messaging REST API cuando todas las conexiones están en uso. También puede elegir qué contexto de usuario se utiliza para la autorización cuando utiliza messaging REST API para enviar, recibir, examinar o publicar un mensaje.

Procedimiento

- "Habilitación del messaging REST API" en la página 869
- "Configuración de la agrupación de conexiones para messaging REST API" en la página 870
- V 9.4.0
  "Configuración del contexto de usuario que se utiliza para la autorización en messaging REST API" en la página 874

Habilitación del messaging REST API

Puede configurar si el messaging REST API está habilitado utilizando el mandato **setmqweb** . De forma predeterminada, messaging REST API está habilitado.

Antes de empezar

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.
- V 9.4.0 Linux Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Procedimiento

- Visualice la configuración actual del messaging REST API utilizando el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

  El campo `mqRestMessagingEnabled` muestra si está habilitada la messaging REST API. Si el valor es `True` , messaging REST API está habilitado.
- Habilite messaging REST API utilizando el mandato siguiente:

  ```
  setmqweb properties -k mqRestMessagingEnabled -v true
  ```
- Inhabilite messaging REST API utilizando el mandato siguiente:

  ```
  setmqweb properties -k mqRestMessagingEnabled -v false
  ```

Tareas relacionadas

“Configuración de la agrupación de conexiones para messaging REST API” en la página 870
Puede configurar el número máximo de conexiones agrupadas que puede utilizar el messaging REST API, y el comportamiento del messaging REST API cuando todas las conexiones están en uso.

“Configuración del contexto de usuario que se utiliza para la autorización en messaging REST API” en la página 874
V 9.4.0 Puede configurar qué contexto de usuario se utiliza para la autorización cuando utiliza messaging REST API para enviar, recibir, examinar o publicar un mensaje. Es decir, puede elegir si el usuario que ha iniciado la sesión en messaging REST API, o el usuario que ha iniciado el servidor mqweb, se utiliza para la autorización.

“Configuración de la modalidad de conexión para messaging REST API” en la página 872
Puede configurar messaging REST API para conectarse a gestores de colas locales o remotos.

Configuración de la agrupación de conexiones para messaging REST API

Puede configurar el número máximo de conexiones agrupadas que puede utilizar el messaging REST API, y el comportamiento del messaging REST API cuando todas las conexiones están en uso.

Antes de empezar

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.
- V 9.4.0 Linux Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Acerca de esta tarea

Para optimizar el rendimiento de messaging REST API, se han agrupado las conexiones con los gestores de colas de IBM MQ. Es decir, en lugar de que cada solicitud REST cree, utilice y suprima su propia conexión, cada solicitud REST utiliza una conexión de una agrupación de conexiones. De forma predeterminada, hay 20 conexiones disponibles para cada agrupación de gestores de colas, y puede elegir entre tres opciones para manejar solicitudes cuando todas las conexiones están en uso:

- El messaging REST API puede crear una nueva conexión no agrupada para utilizarla para la solicitud. Este es el comportamiento predeterminado.
- La messaging REST API puede devolver un error.
- La messaging REST API puede esperar a que una conexión agrupada se vuelva disponible. Esta espera es indefinida.

Puede cambiar el número máximo de conexiones agrupadas y el comportamiento predeterminado de la messaging REST API cuando todas las conexiones están en uso utilizando el mandato **setmqweb properties**.

Procedimiento

- Visualice la configuración actual utilizando el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

  – El campo mqRestMessagingFullPoolBehavior muestra el comportamiento de messaging REST API cuando todas las conexiones dentro de la agrupación están en uso. Si el valor es block, el messaging REST API debe esperar a que haya una conexión disponible. Si el valor es error, el messaging REST API debe devolver un error. Si el valor es overflow, el messaging REST API debe crear una conexión no agrupada para utilizarla y desechar la conexión después de utilizarla.
  – El campo mqRestMessagingMaxPoolSize muestra el tamaño máximo de la agrupación de conexiones.
- Configure el comportamiento de la messaging REST API cuando todas las conexiones dentro de la agrupación están en uso utilizando el mandato siguiente:

  ```
  setmqweb properties -k mqRestMessagingFullPoolBehavior -v action
  ```

  donde *acción* especifica la acción que se debe llevar a cabo. *acción* puede ser uno de los valores siguientes:

  bloque
  Cuando estén en uso todas las conexiones de la agrupación, se espera a que haya una conexión disponible.

  error
  Cuando estén en uso todas las conexiones de la agrupación, se devuelve un error.

desbordamiento
Cuando todas las conexiones de la agrupación estén en uso, cree una conexión no agrupada para utilizarla y deseche la conexión después de utilizarla.

- Configure el tamaño máximo de agrupación de conexiones para cada agrupación de gestores de colas utilizando el mandato siguiente:

```
setmqweb properties -k mqRestMessagingMaxPoolSize -v size
```

donde *tamaño* especifica el tamaño de la agrupación.

Nota: Si se establece un valor grande para *mqRestMessagingMaxPoolSize* y se conectan muchos gestores de colas, considere la posibilidad de aumentar el tamaño máximo del almacenamiento dinámico del servidor mqweb. Para obtener más información, consulte ajuste de la JVM del servidor mqweb .

Tareas relacionadas

"Habilitación del messaging REST API" en la página 869
Puede configurar si el messaging REST API está habilitado utilizando el mandato **setmqweb** . De forma predeterminada, messaging REST API está habilitado.

"Configuración del contexto de usuario que se utiliza para la autorización en messaging REST API" en la página 874
V 9.4.0 Puede configurar qué contexto de usuario se utiliza para la autorización cuando utiliza messaging REST API para enviar, recibir, examinar o publicar un mensaje. Es decir, puede elegir si el usuario que ha iniciado la sesión en messaging REST API, o el usuario que ha iniciado el servidor mqweb, se utiliza para la autorización.

"Configuración de la modalidad de conexión para messaging REST API" en la página 872
Puede configurar messaging REST API para conectarse a gestores de colas locales o remotos.

V 9.4.0 Configuración de la modalidad de conexión para messaging REST API

Puede configurar messaging REST API para conectarse a gestores de colas locales o remotos.

Antes de empezar

Nota: Linux V 9.4.0 Si el servidor mqweb se ejecuta en una instalación autónoma de IBM MQ Web Server , esta tarea no es aplicable. El IBM MQ Web Server autónomo sólo da soporte a conexiones con gestores de colas remotos.

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Acerca de esta tarea

La modalidad de conexión predeterminada para messaging REST API varía en función del tipo de instalación que ejecuta el servidor mqweb:

- En una instalación de IBM MQ , de forma predeterminada, messaging REST API sólo se conecta a gestores de colas locales en la misma instalación que el servidor mqweb. Complete los pasos de esta tarea para ver y cambiar la configuración de conexión.
- Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server , el messaging REST API sólo da soporte a conexiones con gestores de colas remotos. La configuración de conexión no se puede visualizar ni cambiar.

Procedimiento

- Visualice la configuración actual del messaging REST API utilizando el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

  El campo `mqRestMessagingConnectionMode` muestra la modalidad de conexión actual. Si el valor es `local`, el messaging REST API sólo se puede conectar a los gestores de colas en la misma instalación que el servidor mqweb. Si el valor es `remote`, el messaging REST API puede conectarse a gestores de colas remotos.
- Configure el servidor mqweb para permitir que messaging REST API se conecte solo a los gestores de colas que están en la misma instalación que el servidor mqweb utilizando los mandatos siguientes:

  ```
  setmqweb properties -k mqRestMessagingConnectionMode -v local
  endmqweb
  strmqweb
  ```
- Configure el servidor mqweb para permitir que el messaging REST API se conecte a gestores de colas remotos utilizando el mandato siguiente:

  ```
  setmqweb properties -k mqRestMessagingConnectionMode -v remote
  endmqweb
  strmqweb
  ```

Qué hacer a continuación

Si configura el servidor mqweb para permitir que messaging REST API se conecte a gestores de colas remotos, debe proporcionar información de conexión para cada gestor de colas al que desee conectarse. Para obtener más información sobre cómo proporcionar la información de conexión, consulte Configuración de un gestor de colas remoto para su uso con messaging REST API.

Tareas relacionadas

“Habilitación del messaging REST API” en la página 869
Puede configurar si el messaging REST API está habilitado utilizando el mandato **setmqweb** . De forma predeterminada, messaging REST API está habilitado.

“Configuración de la agrupación de conexiones para messaging REST API” en la página 870
Puede configurar el número máximo de conexiones agrupadas que puede utilizar el messaging REST API, y el comportamiento del messaging REST API cuando todas las conexiones están en uso.

“Configuración del contexto de usuario que se utiliza para la autorización en messaging REST API” en la página 874
V 9.4.0 Puede configurar qué contexto de usuario se utiliza para la autorización cuando utiliza messaging REST API para enviar, recibir, examinar o publicar un mensaje. Es decir, puede elegir si el usuario que ha iniciado la sesión en messaging REST API, o el usuario que ha iniciado el servidor mqweb, se utiliza para la autorización.

V 9.4.0 Configuración del contexto de usuario que se utiliza para la autorización en messaging REST API

V 9.4.0 Puede configurar qué contexto de usuario se utiliza para la autorización cuando utiliza messaging REST API para enviar, recibir, examinar o publicar un mensaje. Es decir, puede elegir si el usuario que ha iniciado la sesión en messaging REST API, o el usuario que ha iniciado el servidor mqweb, se utiliza para la autorización.

Antes de empezar

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.
- V 9.4.0 Linux Si el servidor mqweb forma parte de una instalación autónoma de IBM MQ Web Server , debe tener acceso de escritura al archivo `mqwebuser.xml` en el directorio de datos IBM MQ Web Server .

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Acerca de esta tarea

- Cuando el ID de usuario que se utiliza es el ID de usuario que ha iniciado sesión en messaging REST API, **MQMD.UserIdentifier** se establece en el ID de usuario que ha iniciado sesión en la API REST. **MQMD.AppIdentityData** se establece en el ID de usuario que ha iniciado la sesión en la API REST.
- Cuando el ID de usuario que se utiliza es el ID de usuario que ha iniciado el servidor mqweb, **MQMD.UserIdentifier** se deja en blanco. **MQMD.AppIdentityData** se establece en el ID de usuario que ha iniciado la sesión en la API REST.

Consulte MQMD para obtener más información sobre las partes del descriptor de mensaje del mensaje IBM MQ .

Procedimiento

- Visualice la configuración actual del messaging REST API utilizando el mandato siguiente:

  ```
  dspmqweb properties -a
  ```

  El campo `mqRestMessagingAdoptWebUserContext` muestra qué ID de usuario se utiliza para la autorización al enviar, publicar, recibir o examinar mensajes. Si el valor es `True`, el usuario que ha iniciado sesión en messaging REST API se utiliza para la autorización. Si el valor es `False` , el usuario que ha iniciado el servidor mqweb se utiliza para la autorización.

- Configure el messaging REST API para utilizar el ID de usuario del usuario que ha iniciado la sesión en el messaging REST API para la autorización utilizando el mandato siguiente:

  ```
  setmqweb properties -k mqRestMessagingAdoptWebUserContext -v true
  ```

  Cuando **mqRestMessagingAdoptWebUserContext** se establece en `true`, **MQMD.UserIdentifier** se establece en el ID de usuario que ha iniciado la sesión en la API REST. **MQMD.AppIdentityData** se establece en el ID de usuario que ha iniciado la sesión en la API REST.
- Configure el messaging REST API para utilizar el ID de usuario del usuario que ha iniciado el servidor mqweb utilizando el mandato siguiente:

  ```
  setmqweb properties -k mqRestMessagingAdoptWebUserContext -v false
  ```

  Cuando **mqRestMessagingAdoptWebUserContext** se establece en `false`, **MQMD.UserIdentifier** se deja en blanco. **MQMD.AppIdentityData** se establece en el ID de usuario que ha iniciado la sesión en la API REST.

Tareas relacionadas

"Habilitación del messaging REST API" en la página 869
Puede configurar si el messaging REST API está habilitado utilizando el mandato **setmqweb** . De forma predeterminada, messaging REST API está habilitado.

"Configuración de la agrupación de conexiones para messaging REST API" en la página 870
Puede configurar el número máximo de conexiones agrupadas que puede utilizar el messaging REST API, y el comportamiento del messaging REST API cuando todas las conexiones están en uso.

"Configuración de la modalidad de conexión para messaging REST API" en la página 872
Puede configurar messaging REST API para conectarse a gestores de colas locales o remotos.

Configuración de la REST API para MFT

De forma predeterminada, la REST API para MFT no está habilitada. Puede configurar si el REST API para MFT está habilitado, establecer el gestor de colas de coordinación, establecer el gestor de colas de mandatos y especificar el tiempo de espera de reconexión de MFT utilizando el mandato **setmqweb properties** .

Procedimiento

- "Habilitación de REST API para MFT" en la página 875
- "Configuración del gestor de colas de coordinación para REST API para MFT" en la página 876
- "Configuración del gestor de colas de mandatos para REST API para MFT" en la página 878
- "Configuración de los valores de tiempo de espera de REST API para MFT" en la página 879

Habilitación de REST API para MFT

Para poder utilizar REST API para MFT, primero debe habilitar REST API para MFT. Puede configurar si el REST API para MFT está habilitado utilizando el mandato **setmqweb** . De forma predeterminada, la REST API para MFT no está habilitada.

Antes de empezar

Nota: Linux V 9.4.0 Si el servidor mqweb se ejecuta en una instalación autónoma de IBM MQ Web Server , esta tarea no es aplicable. REST API for MFT no está disponible en una instalación autónoma de IBM MQ Web Server .

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Procedimiento

1. Consulte la configuración actual de la REST API para MFT utilizando el mandato siguiente:

   ```
   dspmqweb properties -a
   ```

   El campo `mqRestMftEnabled` muestra si la REST API para MFT está habilitada. El valor es `True` si REST API para MFT está habilitado o `False` de lo contrario.
2. Habilite o inhabilite REST API para MFT utilizando uno de los mandatos siguientes:
   - Habilite REST API para MFT utilizando el mandato siguiente:

     ```
     setmqweb properties -k mqRestMftEnabled -v true
     ```
   - Inhabilite REST API para MFT utilizando el mandato siguiente:

     ```
     setmqweb properties -k mqRestMftEnabled -v false
     ```
3. Reinicie el servidor mqweb especificando los mandatos siguientes:

   ```
   endmqweb
   strmqweb
   ```

Qué hacer a continuación

Si ha habilitado REST API para MFT, debe establecer el nombre del gestor de colas de coordinación para poder utilizar REST API para MFT. Para obtener más información sobre cómo establecer el gestor de colas de coordinación, consulte “Configuración del gestor de colas de coordinación para REST API para MFT” en la página 876.

Configuración del gestor de colas de coordinación para REST API para MFT

Para poder utilizar REST API para MFT, debe configurar un gestor de colas para que actúe como gestor de colas de coordinación para las transacciones MFT . Puede establecer qué gestor de colas es el gestor de colas de coordinación utilizando el mandato **setmqweb** .

Antes de empezar

Nota: Linux V 9.4.0 Si el servidor mqweb se ejecuta en una instalación autónoma de IBM MQ Web Server , esta tarea no es aplicable. REST API for MFT no está disponible en una instalación autónoma de IBM MQ Web Server .

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Procedimiento

1. Consulte la configuración actual de la REST API para MFT utilizando el mandato siguiente:

   ```
   dspmqweb properties -a
   ```

   El campo `mqRestMftCoordinationQmgr` muestra el nombre del gestor de colas de coordinación.
2. Configure el gestor de colas de coordinación utilizando el mandato siguiente:

   ```
   setmqweb properties -k mqRestMftCoordinationQmgr -v qmgrName
   ```

   donde *NombreGestorColas* es el nombre del gestor de colas de coordinación. El gestor de colas de coordinación debe estar en la máquina donde se está ejecutando el servidor mqweb. De forma predeterminada, este nombre de gestor de colas está en blanco. Si un valor no está establecido, la REST API para MFT no funciona.
3. Reinicie el servidor mqweb especificando los mandatos siguientes:

   ```
   endmqweb
   strmqweb
   ```

Qué hacer a continuación

- Asegúrese de que el REST API para MFT esté habilitado. Para obtener más información, consulte “Habilitación de REST API para MFT” en la página 875.
- Si desea utilizar REST API for MFT para enviar solicitudes de creación, debe establecer el nombre del gestor de colas de mandatos. Por ejemplo, si desea utilizar un mandato REST API como, por ejemplo, **create transfer**, debe establecer el nombre del gestor de colas de mandatos. Para obtener más información, consulte “Configuración del gestor de colas de mandatos para REST API para MFT” en la página 878.
- Puede configurar los valores de tiempo de espera de REST API para MFT . El tiempo de espera predeterminado es de 30 minutos. Para obtener más información, consulte “Configuración de los valores de tiempo de espera de REST API para MFT” en la página 879.
- Para utilizar REST API para MFT, un usuario debe estar autenticado en el servidor mqweb y debe ser miembro de uno o varios de los roles `MFTWebAdmino` `MFTWebAdminRO` . Para obtener más información sobre la configuración de usuarios, consulte Configuración de usuarios y roles para REST API.

Configuración del gestor de colas de mandatos para REST API para MFT

Para poder utilizar REST API for MFT para enviar solicitudes de creación, debe establecer el nombre del gestor de colas de mandatos. Por ejemplo, para utilizar el recurso **create transfer** , debe establecer el nombre del gestor de colas de mandatos. Puede establecer el nombre del gestor de colas de mandatos utilizando el mandato **setmqweb** .

Antes de empezar

Nota: Linux V 9.4.0 Si el servidor mqweb se ejecuta en una instalación autónoma de IBM MQ Web Server , esta tarea no es aplicable. REST API for MFT no está disponible en una instalación autónoma de IBM MQ Web Server .

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Procedimiento

1. Consulte la configuración actual de la REST API para MFT utilizando el mandato siguiente:

   ```
   dspmqweb properties -a
   ```

   El campo `mqRestMftCommandQmgr` muestra el nombre del gestor de colas de mandatos.
2. Configure el gestor de colas de mandatos utilizando el mandato siguiente:

   ```
   setmqweb properties -k mqRestMftCommandQmgr -v qmgrName
   ```

   donde *NombreGestorColas* es el nombre del gestor de colas de mandatos. El gestor de colas de mandatos debe estar en la máquina en la que se ejecuta el servidor mqweb. De forma predeterminada, este nombre de gestor de colas está en blanco. Si no se establece un valor, el REST API para MFT para un mandato create no funciona.
3. Reinicie el servidor mqweb especificando los mandatos siguientes:

   ```
   endmqweb
   strmqweb
   ```

Qué hacer a continuación

- Asegúrese de que el REST API para MFT esté habilitado. Para obtener más información, consulte “Habilitación de REST API para MFT” en la página 875.

- Asegúrese de que se ha establecido un gestor de colas de coordinación. Para obtener más información, consulte “Configuración del gestor de colas de coordinación para REST API para MFT” en la página 876.
- Puede configurar los valores de tiempo de espera de REST API para MFT . El tiempo de espera predeterminado es de 30 minutos. Para obtener más información, consulte “Configuración de los valores de tiempo de espera de REST API para MFT” en la página 879.
- Para utilizar REST API para MFT, un usuario debe estar autenticado en el servidor mqweb y debe ser miembro de uno o varios de los roles `MFTWebAdmino` `MFTWebAdminRO` . Para obtener más información sobre la configuración de usuarios, consulte Configuración de usuarios y roles para REST API.

Configuración de los valores de tiempo de espera de REST API para MFT

Puede configurar el periodo de tiempo, en minutos, después del cual REST API for MFT deja de intentar conectarse al gestor de colas de coordinación después de que se interrumpa la conexión. El tiempo de espera predeterminado es de 30 minutos. Puede configurar este tiempo de espera utilizando el mandato **setmqweb** .

Antes de empezar

Nota: Linux V 9.4.0 Si el servidor mqweb se ejecuta en una instalación autónoma de IBM MQ Web Server , esta tarea no es aplicable. REST API for MFT no está disponible en una instalación autónoma de IBM MQ Web Server .

Para completar esta tarea, debe ser un usuario con determinados privilegios para que pueda utilizar los mandatos **dspmqweb** y **setmqweb**:

- z/OS En z/OS, debe tener autorización para ejecutar los mandatos **dspmqweb** y **setmqweb** y acceso de escritura al archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

Atención:

Antes de emitir los mandatos **setmqweb** o **dspmqweb** en z/OS, debe establecer la variable de entorno WLP_USER_DIR para que la variable apunte a la configuración del servidor mqweb.

Para establecer la variable de entorno WLP_USER_DIR, emita 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 Crear el servidor mqweb.

Acerca de esta tarea

Puede configurar el tiempo de espera para REST API para MFT.

REST API for MFT intenta volver a establecer la conexión inmediatamente después de que se interrumpa la conexión con el gestor de colas de coordinación. Si este intento falla, hay un intervalo de cinco minutos entre cada intento de reconexión, hasta que haya transcurrido el tiempo de espera. Por lo tanto, establecer un valor entre 0-5 da como resultado un solo intento de reconexión.

Una vez que la reconexión excede el tiempo de espera, el siguiente intento de reconexión se realiza cuando se invoca alguno de los recursos de la REST API para MFT. Si este intento de reconexión falla, MFT vuelve a intentar reconectarse cada cinco minutos hasta que haya pasado el tiempo de espera de reconexión.

Procedimiento

1. Consulte la configuración actual de la REST API para MFT utilizando el mandato siguiente:

   ```
   dspmqweb properties -a
   ```

   El campo `mqRestMftReconnectTimeoutInMinutes` muestra el valor de tiempo de espera de reconexión, hasta que los servicios REST de transferencia de MFT dejan de intentar conectarse al gestor de colas de coordinación.
2. Configure el tiempo de espera, en minutos, después del cual la REST API para MFT deja de intentar conectarse al gestor de colas de coordinación:
   - Restablezca el tiempo de espera en el valor predeterminado de 30 minutos:

     ```
     setmqweb properties -k mqRestMftReconnectTimeoutInMinutes -d
     ```

   - Establezca el tiempo de espera.

     ```
     setmqweb properties -k mqRestMftReconnectTimeoutInMinutes -v time
     ```

     donde *tiempo* especifica el tiempo, en minutos, antes de que se produzca el tiempo de espera excedido.

     Si este valor está establecido entre `0-5`, la REST API para MFT intenta reconectarse al gestor de colas de coordinación solo una vez. Si la conexión falla, no hay ningún intento de volver a establecer la conexión, hasta que se invoque a la REST API.

     Si este valor está establecido en `-1`, la REST API para MFT intenta reconectarse hasta que la conexión se realice correctamente.
3. Reinicie el servidor mqweb especificando los mandatos siguientes:

   ```
   endmqweb
   strmqweb
   ```

Qué hacer a continuación

- Asegúrese de que el REST API para MFT esté habilitado. Para obtener más información, consulte “Habilitación de REST API para MFT” en la página 875.
- Asegúrese de que se ha establecido un gestor de colas de coordinación. Para obtener más información, consulte “Configuración del gestor de colas de coordinación para REST API para MFT” en la página 876.
- Si desea utilizar REST API for MFT para enviar solicitudes de creación, debe establecer el nombre del gestor de colas de mandatos. Por ejemplo, si desea utilizar un mandato REST API como, por ejemplo, **`create transfer`**, debe establecer el nombre del gestor de colas de mandatos. Para obtener más información, consulte “Configuración del gestor de colas de mandatos para REST API para MFT” en la página 878.
- Para utilizar REST API para MFT, un usuario debe estar autenticado en el servidor mqweb y debe ser miembro de uno o varios de los roles `MFTWebAdmino` `MFTWebAdminRO` . Para obtener más información sobre la configuración de usuarios, consulte Configuración de usuarios y roles para REST API.

Ajuste de la JVM del servidor mqweb

De forma predeterminada, el servidor mqweb Java Virtual Machine (JVM) utiliza valores predeterminados específicos de la plataforma para parámetros de configuración como, por ejemplo, el tamaño mínimo y máximo del almacenamiento dinámico y el tamaño de la memoria caché de clase.

Acerca de esta tarea

Es posible que tenga que cambiar los valores predeterminados para mejorar el rendimiento o para resolver problemas. Por ejemplo, si el servidor mqweb genera un `java.lang.OutOfMemoryError`,

deberá aumentar el tamaño máximo del almacenamiento dinámico. También debe aumentar el tamaño del almacenamiento dinámico si está intentando cargar un gran número de objetos de cola.

Si está experimentando problemas con la visualización de la información de configuración del panel de control en IBM MQ Console, debe establecer una variable que determine la codificación de archivo de la configuración. Puede cambiar los valores predeterminados en el archivo `jvm.options`.

Procedimiento

1. Abra el archivo `jvm.options`.

   El archivo `jvm.options` se encuentra en uno de los siguientes directorios:

   - En una instalación de IBM MQ :
     - Linux AIX En AIX o Linux: `/var/mqm/web/installations/`*`installationName`*`/servers/mqweb`
     - Windows En Windows: *MQ_DATA_PATH*`\web\installations\`*`installationName`*`\servers\mqweb`, donde *MQ_DATA_PATH* es la vía de acceso de datos de IBM MQ . Esta vía de acceso es la vía de acceso de datos que se selecciona durante la instalación de IBM MQ. De forma predeterminada, esta vía de acceso es `C:\ProgramData\IBM\MQ`.
     - IBM i En IBM i: *MQ_DATA_PATH*`/web/installations/Installation1/`
     - z/OS En z/OS: *WLP_user_directory*`/servers/mqweb`

       donde *directorio_usuario_WLP* es el directorio que se ha especificado cuando se ejecutó el script **crtmqweb** para crear la definición del servidor mqweb.
   - Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server : *MQ_OVERRIDE_DATA_PATH*`/web/installations/MQWEBINST/servers/mqweb`

     donde *MQ_OVERRIDE_DATA_PATH* es el directorio de datos de IBM MQ Web Server al que apunta la variable de entorno **MQ_OVERRIDE_DATA_PATH** .
2. Opcional: Establezca el tamaño máximo de almacenamiento dinámico añadiendo la línea siguiente al archivo:

   ```
   -Xmx MaxSize m
   ```

   Donde *tamañoMáx* especifica el tamaño máximo del almacenamiento dinámico, en MB.

   Por ejemplo, la siguiente línea establece el tamaño máximo de almacenamiento dinámico en 1 GB:

   ```
   -Xmx1024m
   ```
3. Opcional: Establezca el tamaño mínimo de almacenamiento dinámico añadiendo la línea siguiente al archivo:

   ```
   -Xms MinSize m
   ```

   Donde *tamañoMín* especifica el tamaño mínimo del almacenamiento dinámico, en MB. Aumentar el tamaño mínimo de almacenamiento dinámico respecto al valor predeterminado puede reducir el tiempo que se tarda en iniciar el servidor mqweb.

   Por ejemplo, la siguiente línea establece el tamaño mínimo de almacenamiento dinámico en 512 MB:

   ```
   -Xms512m
   ```
4. Opcional: Establezca el tamaño de la memoria caché de clase añadiendo la línea siguiente al archivo:

   ```
   -Xscmx Size m
   ```

   Donde *tamaño* especifica el tamaño de la memoria caché de clase, en MB.

Por ejemplo, la línea siguiente establece el tamaño de la memoria caché de clase en 100 MB:

```
-Xscmx100m
```

La memoria caché de clase compartida de Java se utiliza para almacenar datos como, por ejemplo, clases cargadas y código compilado por adelantado (AOT).

La memoria caché de clase reduce de forma significativa el tiempo invertido en iniciar el servidor mqweb. La primera vez que se inicia el servidor mqweb, se crea la memoria caché de clase y el servidor puede tardar un tiempo significativo en iniciarse. Los reinicios posteriores del servidor serán mucho más rápidos, ya que las clases se pueden cargar desde la memoria caché de clase compartida.

Aumentar el tamaño de la memoria caché de clase respecto al valor predeterminado puede reducir el tiempo que se tarda en iniciar el servidor mqweb.

z/OS La memoria caché de clase se vuelve a crear cuando se inicia el servidor mqweb en un sistema z/OS diferente. Por lo tanto, iniciar el servidor mqweb en un sistema z/OS diferente en un sysplex puede tardar un tiempo significativamente más grande que reiniciar el servidor en el mismo sistema.

Tenga en cuenta que los cambios en este valor solo entran en vigor cuando se crea la memoria caché de clase. La memoria caché de clase se crea cuando se inicia el servidor mqweb por primera vez, o después de que la memoria caché de clase se haya destruido utilizando el programa de utilidad de memoria caché de clase Java.

5. Necesario: Compruebe que el archivo contiene las líneas siguientes para especificar la codificación de archivo que se utiliza cuando REST API procesa los datos y para la información de configuración del panel de control del usuario en IBM MQ Console:

```
-Dfile.encoding=UTF-8
-Ddefault.client.encoding=UTF-8
```

6. Reinicie el servidor mqweb.

   z/OS En z/OS, detenga y reinicie la tarea iniciada del servidor mqweb.

   Multi En todas las demás plataformas, especifique los mandatos siguientes en la línea de mandatos:

```
endmqweb
strmqweb
```

Estructura de archivos del componente de instalación de IBM MQ Console y REST API

Existen dos conjuntos de estructuras de directorio que están asociadas con el componente de instalación de IBM MQ Console y REST API. Una estructura de directorio contiene archivos que pueden editarse. La otra estructura de directorio contiene archivos que no pueden editarse.

Archivos editables

Los archivos editables de usuario se establecen como parte de la instalación inicial del componente de instalación de IBM MQ Console y REST API. Dado que estos archivos pueden editarse, los archivos no cambian cuando se aplica mantenimiento.

La ubicación de los archivos editables del usuario depende del sistema operativo y del producto que esté instalado.

- En una instalación de IBM MQ , los archivos editables de usuario se encuentran en uno de los directorios siguientes:
  - Linux AIX En AIX o Linux: `/var/mqm/web/installations/`*`installationName`*

  - Windows En Windows: *MQ_DATA_PATH*\web\installations*installationName*, donde *MQ_DATA_PATH* es la vía de acceso de datos de IBM MQ . Esta vía de acceso es la vía de acceso de datos que se selecciona durante la instalación de IBM MQ. De forma predeterminada, esta vía de acceso es C:\ProgramData\IBM\MQ.
  - z/OS En z/OS: el directorio que se especificó cuando se ejecutó el script **crtmqweb** para crear la definición del servidor mqweb.
- Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server : *MQ_OVERRIDE_DATA_PATH*/web/installations/MQWEBINST

  donde *MQ_OVERRIDE_DATA_PATH* es el directorio de datos de IBM MQ Web Server al que apunta la variable de entorno **MQ_OVERRIDE_DATA_PATH** .

En este directorio de nivel superior, están presentes los siguientes directorios y archivos:

Directorios y archivos	Descripción
angular.persistence/	Directorio donde se almacena la configuración de panel de instrumentos de IBM MQ Console.
servers/	Directorio de servidores WebSphere Liberty .
servers/mqweb	Directorio que contiene la estructura de directorios del servidor mqweb.
servers/mqweb/logs	Directorio que contiene registros para el servidor mqweb.
servers/mqweb/logs/console.log	Registro de estado de servidor básico y mensajes de operación.
servers/mqweb/logs/ffdc	Directorio de salida de Captura de datos en primer error (FFDC).
servers/mqweb/logs/messages.log	Registro de mensajes de tiempo de ejecución del servidor mqweb, incluidos IBM MQ Console y REST API. Los mensajes más antiguos se almacenan en archivos denominados messages_*timestamp*.log.
servers/mqweb/logs/trace.log	Registro de rastreo del servidor mqweb, incluidos IBM MQ Console y REST API. El rastreo más antiguo se almacena en archivos denominados trace_*timestamp*.log. Estos archivos sólo existen si el rastreo está habilitado.
servers/mqweb/logs/state	Estado específico de servidor.
servers/mqweb/server.xml	Archivo de configuración de servidor principal. Este archivo es de solo lectura. Edite el archivo mqwebuser.xml para alterar temporalmente la configuración predeterminada.
servers/mqweb/mqwebuser.xml	Archivo de configuración para IBM MQ Console y REST API. Los valores que están configurados en este archivo alteran temporalmente la configuración predeterminada. Debe ser un usuario privilegiado para poder editar este archivo.

Directorios y archivos	Descripción
`servers/mqweb/resources`	Directorio que contiene diversos recursos de servidor como almacenes de claves.
`servers/mqweb/workarea`	Directorio creado por el servidor mientras opera. Este directorio se crea después de que el servidor se haya ejecutado por primera vez.

Archivos no editables

Los archivos no editables se establecen como parte de la instalación inicial del componente de instalación de IBM MQ Console y REST API. Estos archivos se actualizan cuando se aplica el mantenimiento.

La ubicación de los archivos no editables depende del sistema operativo y del producto que esté instalado.

- En una instalación de IBM MQ , los archivos no editables se encuentran en uno de los directorios siguientes:
  - ALW En AIX, Linux, and Windows: *MQ_INSTALLATION_PATH*`/web`
  - IBM i En IBM i: *MQ_INSTALLATION_PATH*`/web`
  - z/OS En z/OS: *`installation_directory`*`/web/`

    donde *directorio_instalación* es la vía de instalación de IBM MQ for z/OS UNIX System Services Components.
- Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server , el directorio en el que se ha descomprimido el archivo de instalación IBM MQ Web Server .

Están presentes la estructura de directorios y los archivos siguientes en esta ubicación:

Directorios y archivos	Descripción
`bin/`	Directorio que contiene mandatos WebSphere Liberty . Debe ser un usuario privilegiado para poder ejecutar scripts en este directorio.
`mq/`	Estructura de directorios que contiene diversos recursos de IBM MQ.
`mq/apps/`	Directorio que contiene las aplicaciones IBM MQ Console y REST API.
`mq/etc/`	
`mq/etc/mqweb.xml`	Archivo de configuración de sólo lectura para el servidor mqweb. Edite el archivo `mqwebuser.xml` para realizar cambios de configuración.
`mq/libs`	Directorio que contiene bibliotecas compartidas para uso por parte de IBM MQ Console y REST API.
`mq/samp`	Directorio que contiene ejemplos.

Directorios y archivos	Descripción
`mq/samp/configuration`	Directorio que contiene archivos de configuración de ejemplo que se pueden copiar en el archivo `mqwebuser.xml`.

Copia de seguridad y restauración de la configuración del servidor mqweb

Puede realizar una copia de seguridad de la configuración del servidor mqweb y restaurarla en la misma ubicación o en una ubicación diferente.

Antes de empezar

Para poder restaurar la configuración del servidor mqweb, debe instalar IBM MQ, o el IBM MQ Web Serverautónomo, en el sistema donde desea restaurar el servidor mqweb. En una instalación autónoma de IBM MQ Web Server , debe crear el servidor mqweb siguiendo los pasos de "Configuración del IBM MQ Web Server autónomo" en la página 854.

Acerca de esta tarea

Siga el procedimiento de esta tarea para realizar una copia de seguridad y restaurar la configuración del servidor mqweb. Si restaura el servidor mqweb en una ubicación diferente, debe actualizar la configuración del servidor mqweb para asegurarse de que las referencias a los archivos son correctas.

V 9.4.0 También puede utilizar este procedimiento para migrar un servidor mqweb que se ejecuta actualmente en una instalación de IBM MQ para que se ejecute en una instalación autónoma de IBM MQ Web Server .

Procedimiento

1. Para realizar una copia de seguridad de la configuración del servidor mqweb, copie todos los archivos del directorio que contiene la configuración del servidor mqweb en la ubicación de copia de seguridad.
   - En una instalación de IBM MQ , copie el contenido del directorio siguiente:
     - Linux AIX En AIX o Linux: `/var/mqm/web/installations/`*`installationName`*
     - Windows En Windows: *MQ_DATA_PATH*`\web\installations\`*installationName*, donde *MQ_DATA_PATH* es la vía de acceso de datos de IBM MQ . Esta vía de acceso es la vía de acceso de datos que se selecciona durante la instalación de IBM MQ. De forma predeterminada, esta vía de acceso es `C:\ProgramData\IBM\MQ`.
     - z/OS En z/OS: el directorio de usuarios de WebSphere Liberty que se ha especificado cuando se ejecutó el script **crtmqweb** para crear la definición de servidor mqweb.
   - Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server , copie el contenido del directorio *MQ_OVERRIDE_DATA_PATH*`/web/installations/MQWEBINST` , donde *MQ_OVERRIDE_DATA_PATH* es el directorio de datos IBM MQ Web Server al que apunta la variable de entorno **MQ_OVERRIDE_DATA_PATH** .
2. Para restaurar la configuración del servidor mqweb, sustituya el contenido del directorio que contiene la configuración del servidor mqweb por los archivos que ha copiado en el paso "1" en la página 885.
   - En una instalación de IBM MQ , sustituya el contenido del directorio siguiente:
     - Linux AIX En AIX o Linux: `/var/mqm/web/installations/`*`installationName`*
     - Windows En Windows: *MQ_DATA_PATH*`\web\installations\`*installationName*, donde *MQ_DATA_PATH* es la vía de acceso de datos de IBM MQ . Esta vía de acceso es la vía de

acceso de datos que se selecciona durante la instalación de IBM MQ. De forma predeterminada, esta vía de acceso es `C:\ProgramData\IBM\MQ`.

  - z/OS En z/OS: el directorio de usuarios de WebSphere Liberty que se ha especificado cuando se ejecutó el script **crtmqweb** para crear la definición de servidor mqweb.

- Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server , sustituya el contenido del directorio *MQ_OVERRIDE_DATA_PATH*`/web/installations/MQWEBINST` , donde *MQ_OVERRIDE_DATA_PATH* es el directorio de datos de IBM MQ Web Server al que apunta la variable de entorno **MQ_OVERRIDE_DATA_PATH** .

3. Establezca la propiedad de los archivos que ha restaurado en el paso “2” en la página 885 para que el ID de usuario del servidor mqweb pueda acceder a los archivos.
4. Si ha restaurado la configuración del servidor mqweb en una ubicación diferente, cambie el valor de las propiedades de la configuración del servidor mqweb que hacen referencia a los archivos del directorio de configuración del servidor mqweb anterior.
   a) Antes de emitir los mandatos **setmqweb** o **dspmqweb** , establezca el entorno para que apunte a la configuración del servidor mqweb.

   - z/OS En z/OS, establezca la variable de entorno **WLP_USER_DIR** para que la variable apunte a la configuración del servidor mqweb, especificando el mandato siguiente:

     ```
     export WLP_USER_DIR=WLP_user_directory
     ```

     donde *directorio_usuario_WLP* es el nombre del directorio que se pasa al mandato **crtmqweb** .

     Si desea más información, consulte Crear el servidor mqweb.

   - Linux V 9.4.0 En una instalación autónoma de IBM MQ Web Server , establezca la variable de entorno **MQ_OVERRIDE_DATA_PATH** en el directorio de datos IBM MQ Web Server .
   - En todos los demás entornos, no es necesario que realice ninguna acción para establecer el entorno.

   b) Ver el valor de todas las propiedades de servidor mqweb configurables que un usuario ha modificado. Emita el mandato siguiente:

   ```
   dspmqweb properties -u
   ```

   c) Si se visualiza la propiedad **remoteKeyfile** , compruebe el valor de la propiedad.

   Si el valor de la propiedad hace referencia a una vía de acceso de archivo en el directorio de configuración del servidor mqweb anterior, cambie el valor para que haga referencia a la vía de acceso de archivo en el nuevo directorio de configuración del servidor mqweb. Emita el mandato siguiente para cambiar el valor de la propiedad **remoteKeyfile** :

   ```
   setmqweb properties -k remoteKeyfile -v path_to_keyfile
   ```

   d) Ver la configuración del gestor de colas remoto del servidor mqweb. Emita el mandato siguiente:

   ```
   dspmqweb remote -a
   ```

   e) Si se visualiza alguna de las propiedades siguientes, compruebe el valor de la propiedad:

   - **globalTrustStorePath**
   - **globalKeyStorePath**
   - **ccdtURL**
   - **keyStorePath**
   - **trustStorePath**

   Cambie el valor de cualquier propiedad que haga referencia a una vía de acceso de archivo en el directorio de configuración del servidor mqweb anterior para que haga referencia a la vía de acceso de archivo en el nuevo directorio de configuración del servidor mqweb. Emita el mandato

setmqweb remote para cambiar el valor de cada propiedad. Por ejemplo, para cambiar el valor de la propiedad **keyStorePath** para el gestor de colas remoto con el nombre exclusivo `remote-QM1`, emita el mandato siguiente:

```
setmqweb remote -uniqueName remote-QM1 -keyStorePath new_keystore_path
```

Para obtener más información, consulte setmqweb remote (set mqweb server remote queue manager configuration).

Windows MQ Adv. Linux MQ Adv. VUE MQ Adv. z/OS Definición de una conexión de Aspera gateway en plataformas Linux o Windows

El IBM Aspera faspio Gateway proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de red para IBM MQ. Un gestor de colas que se ejecuta en cualquier plataforma autorizada puede conectarse a través de un Aspera gateway. La propia pasarela se despliega en Red Hat o Ubuntu Linux, o Windows.

Acerca de esta tarea

Aspera gateway se puede utilizar para mejorar el rendimiento de los canales del gestor de colas. Es especialmente eficaz si la red tiene una latencia alta o tiende a perder paquetes y, normalmente, se utiliza para acelerar la conexión entre gestores de colas en distintos centros de datos.

Nota: Para una red rápida que no pierde paquetes, hay una disminución en el rendimiento cuando se utiliza Aspera gateway, por lo que es importante comprobar el rendimiento de la red antes y después de definir una conexión Aspera gateway.

Defina un Aspera gateway en cada extremo de la conexión de red IP y, a continuación, utilice TCP/IP para conectar canales de gestor de canales a cada pasarela. Un gestor de colas no necesita estar en ejecución en la misma máquina que utiliza Aspera gateway y varios gestores de colas pueden utilizar la misma pasarela.

Para utilizar Aspera gateway, debe tener una o varias de las siguientes titularidades:

- MQ Adv. IBM MQ Advanced for Multiplatforms
- MQ Appliance IBM MQ Appliance
- MQ Adv. VUE IBM MQ Advanced for z/OS VUE
- MQ Adv. z/OS IBM MQ Advanced for z/OS

Puede desplegar Aspera gateway en cualquiera de las plataformas siguientes:

- Linux for x86-64
- Linux on Power Systems - Little Endian
- Linux for IBM Z
- Windows - para obtener más información sobre el soporte de plataforma en Windows, consulte Documentación de IBM Aspera faspio Gateway.

El uso de Aspera gateway se limita a los mensajes de IBM MQ a menos que la pasarela tenga titularidad independiente.

Los gestores de colas que utilizan Aspera gateway se pueden ejecutar en cualquier plataforma soportada. Para obtener una lista completa de las plataformas soportadas, consulte Iconos utilizados en la documentación del producto.

Para cada gestor de colas que no está en la misma máquina que utiliza Aspera gateway, compruebe que tenga una conexión de red rápida entre el gestor de colas y Aspera gateway.

Utilice un archivo `toml` para crear una definición de pasarela que defina los puertos de entrada y salida que utiliza la pasarela. Un archivo `toml` de ejemplo se envía con Aspera gateway. La definición de la

pasarela de salida define la conexión del gestor de colas local con la pasarela y de la pasarela local con la pasarela remota. La definición de pasarela de entrada define la conexión de la pasarela remota con la pasarela local y de la pasarela local con el gestor de colas local.

Los pasos siguientes proporcionan una guía básica para la activación y ejecución. Para obtener información más detallada, consulte la documentación de IBM Aspera faspio Gateway.

Procedimiento

1. Obtenga la imagen de instalación de Aspera gateway.

   Multi En Multiplatforms, puede descargar Aspera gateway desde Passport Advantage. eAssembly de "IBM Aspera faspio Continuous Delivery Release for IBM MQ V9.4 Multiplatform Multilingual eAssembly". Se entrega como una imagen Continuous Delivery (CD) sólo debido al ritmo de cambio en este área, lo que significa que se necesitan actualizaciones en la frecuencia de los releases de CD y puede instalarlo en cualquier sistema IBM MQ que tenga la titularidad IBM MQ Advanced for Multiplatforms o IBM MQ Appliance. Para descargar este eAssembly, vaya a Descarga de IBM MQ 9.4 y, a continuación, pulse la pestaña del release necesario. eAssembly contiene las imágenes de instalación para todas las plataformas en las que está disponible la pasarela. El eAssembly también contiene un archivo `ibm-faspio-license.zip` , que contiene un archivo de licencia.

   MQ Adv. VUE MQ Adv. z/OS Si el sistema IBM MQ tiene titularidad IBM MQ Advanced for z/OS VUE , o titularidad IBM MQ Advanced for z/OS , obtendrá el Aspera gateway del componente de Connector Pack que forma parte de la instalación de SMP/E.

   MQ Adv. VUE MQ Adv. z/OS Los archivos para IBM MQ Advanced for z/OS VUE y IBM MQ Advanced for z/OS son los siguientes:

   Tabla 53. Nombres de archivo y números de versión de faspio por plataforma y versión de IBM MQ

Plataforma	Nombre de archivo	número de versión de faspio
Linux for x86-64	V 9.4.0 V 9.4.0 `M0GVDEN.zip`	1.3.4
Linux on Power Systems - Little Endian	V 9.4.0 V 9.4.0 `M0GVFEN.zip`	1.3.4
Linux for IBM Z	V 9.4.0 V 9.4.0 `M0GVGEN.zip`	1.3.4
Windows	V 9.4.0 V 9.4.0 `M0GVHEN.zip`	1.3.4

   Nota: Aspera gateway no se puede ejecutar de forma nativa en z/OS.

   .

   MQ Adv. VUE MQ Adv. z/OS Además de las imágenes de instalación, el directorio `fasp` contiene `M05QKEN.zip`, que contiene un archivo de licencia.

2. Copie la imagen de instalación de Aspera gateway en las dos máquinas que van a ejecutar la pasarela y, a continuación, extraiga e instale la pasarela.

   Utilice el archivo de licencia contenido en `ibm-faspio-license.zip` (Multiplatforms) o `M05QKEN.zip` (z/OS). Para obtener más información, consulte la documentación de IBM Aspera faspio Gateway :

- Linux Instalación en Linux
- Windows Instalación en Windows

3. Configure y proteja cada pasarela.

   Para obtener más información, consulte la documentación de IBM Aspera faspio Gateway:

   - Configuración del archivo de configuración de pasarela
   - Protección de la pasarela

4. En cada extremo de la conexión de red, cambie la definición de canal para conectarse al puerto en el que escucha la pasarela local.
5. Inicie cada servicio de pasarela.

   Para obtener más información, consulte la documentación de IBM Aspera faspio Gateway:

   - Linux Inicio en Linux
   - Windows Inicio en Windows

6. Reinicie los canales.

   Ahora los gestores de colas se comunican a través de una conexión Aspera gateway.

Ejemplo

Este ejemplo define una conexión Aspera gateway en dos máquinas que ejecutan Linux. a configuración es como sigue:

- La dirección IP de la máquina de pasarela local es 9.20.193.107. La dirección IP de la máquina de pasarela remota es 9.20.192.115.
- El gestor de colas local se está ejecutando en una máquina con la dirección IP 9.20.121.5. El gestor de colas remoto se está ejecutando en una máquina con la dirección IP 9.20.121.25. Ambos gestores de colas están escuchando el puerto 1414.
- El canal del gestor de colas en el gestor de colas local se ha cambiado para conectarse al Aspera gateway local utilizando **conname** 9.20.193.107 (1500). El canal del gestor de colas del gestor de colas remoto se ha cambiado para conectarse al Aspera gateway remoto utilizando **conname** 9.20.192.115 (1500).
- A partir de IBM Aspera faspio Gateway 1.2, TLS está habilitado de forma predeterminada. Si desea configurar TLS con la pasarela, consulte Protección de la pasarela en la documentación de IBM Aspera faspio Gateway.

1. Defina una conexión de Aspera gateway en la máquina de pasarela local:
   - Instale Aspera gateway:
     - Linux En Linux, utilice el mandato siguiente:

       ```
       rpm -ivh ibm-faspio-gateway-<version>.x86_64.rpm
       ```

   - Modifique el archivo `gateway.toml` en el directorio que la instalación ha creado:

     Edite el archivo para establecer las definiciones de pasarela local.

     ```
     [[bridge]]
         name = "Outbound"
         [bridge.local]
                protocol = "tcp"
                    host = "9.20.193.107"
                    port = 1500
         tls_enabled = false

         [bridge.forward]
                protocol = "fasp"
                    host = "9.20.192.115"
                    port = 1600
         tls_enabled = false
     ```

```
[[bridge]]
    name = "Inbound"
    [bridge.local]
           protocol = "fasp"
               host = "9.20.193.107"
               port = 1600
    tls_enabled = false

    [bridge.forward]
           protocol = "tcp"
               host = "9.20.121.5"
               port = 1414
    tls_enabled = false
```

- Copie el archivo `aspera-license` de `ibm-faspio-license.zip` (Multiplatforms) o `M05QKEN.zip` (z/OS) en `/usr/local/etc/faspio/`.

2. Repita el paso anterior para definir una conexión Aspera gateway en la máquina de pasarela remota.
   - Modifique el archivo `gateway.toml` en el directorio que ha creado la instalación. Edite el archivo para establecer las definiciones de pasarela remota:

```
[[bridge]]
    name = "Outbound"
    [bridge.local]
           protocol = "tcp"
               host = "9.20.193.107"
               port = 1500
    tls_enabled = false

    [bridge.forward]
           protocol = "fasp"
               host = "9.20.192.115"
               port = 1600
    tls_enabled = false

[[bridge]]
    name = "Inbound"
    [bridge.local]
           protocol = "fasp"
               host = "9.20.193.107"
               port = 1600
    tls_enabled = false

    [bridge.forward]
           protocol = "tcp"
               host = "9.20.121.5"
               port = 1414
    tls_enabled = false
```

   - Copie el archivo `aspera-license` de `ibm-faspio-license.zip` (Multiplatforms) o `M05QKEN.zip` (z/OS) en `/usr/local/etc/faspio/`.
3. En cada extremo de la conexión, cambie la definición de canal para conectarse al puerto en el que escucha la pasarela local.
   - Cambie el canal del gestor de colas en el gestor de colas local para conectarse al Aspera gateway local utilizando **conname** 9.20.193.107 (1500).
   - Cambie el canal del gestor de colas en el gestor de colas remoto para conectarse al Aspera gateway remoto utilizando **conname** 9.20.192.115 (1500).
4. Inicie la pasarela local ejecutando el mandato siguiente en la máquina de pasarela local:
   - Linux

```
sudo systemctl start faspio-gateway
```

5. Inicie la pasarela remota ejecutando el mandato siguiente en la máquina de pasarela remota:
   - Linux

```
sudo systemctl start faspio-gateway
```

6. Reinicie los canales.

Qué hacer a continuación

Aspera gateway pasa los datos que recibe sin interpretarlos de ninguna forma. Esto significa que puede configurar TLS entre los canales de gestor de colas que están utilizando Aspera gateway porque la conexión de pasarela no tiene conocimiento del reconocimiento TLS. Esto también significa que los gestores de colas en cualquier plataforma IBM MQ soportada pueden utilizar Aspera gateway.

Para utilizar un gestor de colas multiinstancia con la pasarela, configure las definiciones de pasarela para cada instancia del gestor de colas.

Nota: El Aspera gateway sólo se ha probado con canales de gestor de colas. No se ha probado con los canales cliente. Esto se debe a que el uso previsto para Aspera gateway es conectar gestores de colas remotos a través de una red lenta, mientras que las aplicaciones cliente normalmente se conectan a gestores de colas en un centro de datos local a través de una red rápida.

Referencia relacionada

"Qué tipo de comunicación utilizar" en la página 17
Diferentes plataformas dan soporte a diferentes protocolos de comunicación. El protocolo de transmisión que elija dependerá de su combinación de plataformas de servidor y IBM MQ MQI client.

Documentación de IBM Aspera faspio Gateway

Multi Configuración de IBM MQ para su uso con el servicio de calibración de IBM Cloud Private

Configuración de IBM MQ para su uso con el servicio de calibración de IBM Cloud Private para notificar y ver información de inicio y uso del gestor de colas.

Antes de empezar

Antes de configurar gestores de colas IBM MQ para utilizar un servicio IBM Cloud Private, debe tener una cuenta IBM Cloud. Para crear la cuenta, consulte Registro en IBM Cloud.

Acerca de esta tarea

Mediante el uso del servicio de calibración de IBM Cloud Private, puede conectar los productos IBM locales a la instancia del servicio en IBM Cloud Private y ver todos los productos registrados en la organización como un solo panel de instrumentos.

Puede configurar y conectar los gestores de colas de AIX, Linux y Windows a la instancia del servicio de calibración y ver su información de inicio y uso. Sin embargo, en plataformas distintas a los entornos del contenedor Linux, los datos no se pueden utilizar para dar soporte a licencias de determinación de precios basadas en el contenedor por horas.

Para registrar datos de uso para un tipo de licencia de VPC mensual, en lugar de la métrica de licencia por hora predeterminada, establezca la variable de entorno `AMQ_LICENSING_METRIC=VPCMonthlyPeak`. Esto hace que el gestor de colas cargue los datos relacionados con los tipos de licencia de VPC mensuales, en lugar del comportamiento predeterminado de cargar los datos relacionados con las licencias basadas en contenedores por horas.

Utilice los atributos siguientes con la stanza `ReportingService`del archivo `qm.ini`:

APIKeyFile
Ubicación del archivo de texto con el valor **APIKey** de la instancia del servicio de calibración.

CapacityReporting
Escribe mensajes de registro de error periódicamente en los registros AMQERR con el formato siguiente:

```
4/22/2020 01:44:29 PM - Process(1274.1) User(bld-adm) Program(amqmgr0)
```

```
Host(8b3b83f2bc7d) Installation(Docker)
VRMF(9.2.0.0)
Time(2020-04-22T13:44:29.295Z)
ArithInsert1(300)
CommentInsert1(8.5)
CommentInsert2(IBM MQ Advanced)
```

La información generada por el atributo **CapacityReporting** se inserta en el mensaje AMQ5064, que le proporciona una mejor comprensión sobre el grado de uso de IBM MQ que está realizando su empresa:

AMQ5064
Este gestor de colas se ha estado ejecutando durante 300 segundos. Actualmente, se está ejecutando con 8,5 núcleos. El tipo de licencia es IBM MQ Advanced.

Gravedad
0: Información

Explicación
Este es un mensaje informativo para el rastreo del uso.

Respuesta
Ninguna.

LicensingGroup
El grupo de facturación al que pertenece el gestor de colas. Esto afecta a la forma en la que se agrupan los datos en los informes generados por el servicio de calibración.

ServiceURL
La dirección de servicio de IBM Cloud Private.

ServiceProxy
El URL y el puerto para el proxy HTTP que se pueden utilizar si los gestores de colas no tienen acceso directo a la red en la que se está ejecutando el servicio de calibración.

Puede ver los hosts en los que están instalados sus productos, las versiones de producto que está utilizando y las plataformas donde se ejecutan. A partir de las métricas de uso de alto nivel que se muestran para cada producto, puede tener una visión general de cómo son de grandes las cargas de trabajo. Para IBM MQ, puede ver qué gestores de colas se utilizan más y cuáles tienen las cargas de trabajo más ligeras.

Cuando un gestor de colas se ha configurado para conectarse a una instancia del servicio de calibración, se notifica la información siguiente a IBM Cloud Private:

- El nombre del gestor de colas de IBM MQ
- El identificador del gestor de colas de IBM MQ
- El directorio raíz de instalación de IBM MQ
- Los componentes instalados de IBM MQ (nombre y versión)
- Nombre de host
- Nombre del sistema operativo del host
- Versión del sistema operativo del host
- Información de uso del núcleo de procesador virtual (VPC) para el gestor de colas IBM MQ

Puede supervisar las métricas de uso de VPC del gestor de colas en el panel de instrumentos de la instancia del servicio de calibración.

Procedimiento

- Configure un gestor de colas para utilizarlo con la instancia del servicio de calibración en IBM Cloud Private.
- Conéctese al servicio de calibración de IBM Cloud Private a través de un proxy HTTP.
- Resuelva los problemas de la conexión con el servicio de calibración de IBM Cloud Private.

Referencia relacionada

Métrica de precios para núcleos de procesador virtual (VPC)

Multi Configuración de un gestor de colas para utilizarlo con la instancia del servicio de calibración en IBM Cloud Private

Configure la información de seguridad y registro de IBM Cloud para el gestor de colas y, a continuación, conéctese a la instancia del servicio de calibración que ya ha creado.

Acerca de esta tarea

El panel de instrumentos de la instancia del servicio de calibración IBM Cloud Private muestra datos solo para los gestores de colas que se han configurado para incluir la información de seguridad y registro de IBM Cloud Private.

Procedimiento

1. Siga los pasos documentados de IBM Cloud Private para crear un ID de servicio en:

   Creación de un ID de servicio utilizando la CLI de IBM Cloud Private.
2. Siga los pasos documentados de IBM Cloud Private para crear una clave de API en:

   API de gestión de claves de API.
3. Descargue los certificados TLS del clúster IBM Cloud Private .

   Tome nota de la ubicación donde ha descargado los certificados. Puede añadir los certificados descargados al repositorio de claves para el gestor de colas, en el paso “9” en la página 894.
4. Cree un archivo de texto `apikeyfile.txt` y añada el valor **`API key`** que ha copiado en la tarea anterior.

   Tenga en cuenta la ubicación de `apikeyfile.txt` para que pueda incluir la vía de acceso en el Paso 8. Este archivo debe ser legible por el usuario del gestor de colas (*'mqm*' en sistemas AIX and Linux). El archivo sólo debe contener el propio **`API key`** , no una carga útil JSON, por ejemplo `d9c11b45-4dda-4de4-c0b2-2e4e1004dc64`.
5. Cree el gestor de colas, por ejemplo, *QM1*.

   Para obtener más información, consulte Creación y gestión de gestores de colas en Multiplatforms.
6. Inicie el gestor de colas *QM1*.

   Para obtener más información, consulte Inicio de un gestor de colas.
7. Recuerde configurar el entorno de la línea de mandatos de IBM MQ antes de ejecutar los mandatos de IBM MQ.

   Ejecute el mandato **`setmqenv`**.

   AIX En AIX:

   ```
   . /usr/mqm/bin/setmqenv -s
   ```

   Linux En Linux:

   ```
   . /opt/mqm/bin/setmqenv -s
   ```

   Windows En Windows:

   ```
   "C:\Program Files\IBM\MQ\bin\setmqenv.cmd" -n installation name
   ```
8. Cree un almacén de confianza SSL para el gestor de colas *QM1*.

AIX Empiece a crear el almacén de confianza en AIX:

```
runmqakm -keydb -create -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms
-expire 30 -stash
```

Linux En Linux:

```
runmqakm -keydb -create -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms
-expire 30 -stash
```

Windows En Windows:

```
runmqakm -keydb -create -db "MQ data directory\qmgrs\QM1\ssl\key.kdb" -pw password -type
cms -expire 30 -stash
```

9. Añada los certificados digitales que ha descargado en el paso “3” en la página 893al repositorio de claves del gestor de colas.

   AIX En AIX:

   ```
   runmqakm -cert -add -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms
   -label RootCA
            -file Download_location/RootCA.crt -format ascii -trust enable

   runmqakm -cert -add -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms
   -label ServerCert
            -file Download_location/CERT.crt -format ascii -trust enable
   ```

   Linux En Linux:

   ```
   runmqakm -cert -add -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms
   -label RootCA
            -file Download_location/RootCA.crt -format ascii -trust enable

   runmqakm -cert -add -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms
   -label ServerCert
            -file Download_location/CERT.crt -format ascii -trust enable
   ```

   Windows En Windows:

   ```
   runmqakm -cert -add -db "MQ data directory\qmgrs\QM1\ssl\key.kdb" -pw password -type cms
   -label RootCA
            -file "Download_location\RootCA.crt" -format ascii -trust enable

   runmqakm -cert -add -db "C:\ProgramData\IBM\MQ\qmgrs\QM1\ssl\key.kdb" -pw password -type
   cms -label ServerCert
            -file "Download_location\CERT.crt" -format ascii -trust enable
   ```

10. Añada la nueva stanza de `ReportingService` con la vía de acceso `apikeyfile` del archivo `qm.ini` del gestor de colas:

    ```
    ReportingService:
       APIKeyFile=APIKey file location/apikeyfile.txt
    ```

11. Añada el valor **API host** al archivo `qm.ini` .

    La sección de stanza `ReportingService` ahora contiene la vía de acceso a los valores `apikeyfile` y **API host** (**ServiceURL**):

    ```
    ReportingService:
       APIKeyFile=APIKey file location/apikeyfile.txt
       ServiceURL=https://productinsights-api.ng.bluemix.net
    ```

    Guarde y salga del archivo `qm.ini`.

12. Reinicie el gestor de colas para que se apliquen los cambios.

Puede que se le solicite que otorgue permiso al proceso del gestor de colas **amqzmur0** para acceder a la red. El acceso es necesario para permitir que el gestor de colas contacte con el servicio de calibración.

13. Consulte la información sobre el gestor de colas *QM1* en la instancia del servicio de calibración.

    Cuando el estado de creación de informes es activo, la información de inicio y uso para todos los servidores de integración en el nodo de integración especificado se notifica al servicio de calibración. La información de uso se actualiza cada 15 minutos.
14. Opcional: Detenga un gestor de colas para que deje de notificar al servicio de calibración, eliminando la stanza `ReportingService` del archivo `qm.ini` del gestor de colas y reinicie el gestor de colas.
15. Opcional: Consulte la información de diagnósticos en el archivo de registro del gestor de colas si el gestor de colas no puede notificar información de inicio o uso al servicio de calibración.

    AIX En AIX:

    ```
    /var/mqm/qmgrs/QM1/errors/AMQERR0*.log
    ```

    Linux En Linux:

    ```
    /var/mqm/qmgrs/QM1/errors/AMQERR0*.log
    ```

    Windows En Windows:

    ```
    C:\ProgramData\IBM\MQ\errors\AMQERR0*.log
    ```

Resultados

Ha creado una instancia del servicio de calibración y ha configurado el gestor de colas para que se conecte a la instancia. Puede ver la información sobre el gestor de colas en el panel de instrumentos de la instancia del servicio de calibración.

Multi Conexión al servicio de calibración IBM Cloud Private a través de un proxy HTTP

Si el gestor de colas se está ejecutando en un sistema que no tiene acceso directorio al clúster ICP, puede utilizar un proxy HTTP que proporciona la organización para conectarse a la instancia del servicio de calibración en IBM Cloud Private.

Antes de empezar

Ha configurado la seguridad, ha añadido el **API key** y el URL de servicio al archivo `qm.ini` para el gestor de colas.

Acerca de esta tarea

Utilice esta tarea para configurar el gestor de colas para conectarse a la instancia del servicio de calibración en IBM Cloud Private a través de un proxy HTTP proporcionado por la organización.

Procedimiento

- Añada un atributo de proxy de servicio a la stanza de registro de IBM Cloud Private del archivo `qm.ini`.

  Se puede configurar el atributo **ServiceProxy** de la siguiente manera:

  – Un URL que incluya el prefijo http:// y, de forma opcional, el puerto. Si no se especifica el puerto, se utiliza *1080*.

```
ReportingService:
   ServiceProxy=http://myorgproxy.net:1080
```

Nota: El parámetro **ServiceProxy** tiene que estar establecido a un URL http:// válido. Otros protocolos de proxy como, por ejemplo, HTTPS y SOCKS no están soportados.

- Reinicie el gestor de colas antes de que entren en vigor los cambios.

Multi Resolución de problemas de la conexión con el servicio de calibración

Consejos para la resolución de problemas para los errores que puede encontrar al conectar el gestor de colas con una instancia del servicio de calibración.

El gestor de colas no se puede registrar con o cargar las métricas de uso en el servicio de calibración configurado

Compruebe que el gestor de colas tiene acceso a la red. El valor **APIKey** del archivo de claves de API es incorrecto. Asegúrese de que el componente IBM Global Security Kit (GSKit) esté instalado.

Stanza `qm.ini` no válida

Se ha encontrado una stanza `qm.ini` no válida. Busque en el registro cronológico de errores para obtener más información.

Parámetro de servicio proxy HTTP no válido

El valor para el atributo **ServiceProxy** para la stanza `ReportingService` del gestor de colas no se ha configurado correctamente. El gestor de colas no se registra en el servicio. El parámetro **ServiceProxy** tiene que estar establecido a un URL http:// válido. Otros protocolos de proxy como, por ejemplo, HTTPS y SOCKS no están soportados.

z/OS Configuring queue managers on z/OS

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

Before you begin

Before you configure IBM MQ for z/OS, read:

- IBM MQ for z/OS concepts
- Planning your IBM MQ environment on z/OS

About this task

After you have installed IBM MQ, you must carry out a number of tasks before you can make it available to users.

Procedure

- See the following subtopics for information on how to configure queue managers on IBM MQ for z/OS.

Related concepts

z/OS Sources from which you can issue MQSC and PCF commands on IBM MQ for z/OS

Related tasks

“Creación de gestores de colas en Multiplatforms” on page 7
Antes de poder utilizar mensajes y colas, debe crear e iniciar al menos un gestor de colas y los objetos asociados al mismo. Un gestor de colas gestiona los recursos que tiene asociados, en particular las colas

que posee. Proporciona servicios de colocación en cola a las aplicaciones para llamadas y mandatos MQI (Message Queuing Interface) para crear, modificar, mostrar y suprimir objetos de IBM MQ.

Securing

"Configuración de la gestión de colas distribuidas" on page 210
En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

"Configuración de conexiones entre el cliente y el servidor" on page 16
Para configurar los enlaces de comunicación entre IBM MQ MQI clients y servidores, decida el protocolo de comunicación, defina las conexiones en ambos extremos del enlace, inicie un escucha y defina canales.

z/OS Administering IBM MQ for z/OS

Planning

Related reference

z/OS Using the IBM MQ for z/OS utilities

z/OS Preparing to customize queue managers on z/OS

Use this topic when customizing your queue managers with details of installable features, national language features, and information about testing, and setting up security.

Preparing for customization

The Program Directory lists the contents of the IBM MQ installation tape, the program and service level information for IBM MQ, and describes how to install IBM MQ for z/OS using System Modification Program Extended (SMP/E). Para enlaces de descarga de los directorios de programas, consulteIBM MQ for z/OS Archivos PDF del directorio de programas .

When you have installed IBM MQ, you must carry out a number of tasks before you can make it available to users. See the following sections for a description of these tasks:

- "Setting up IBM MQ for z/OS" on page 901
- "Testing a queue manager on z/OS" on page 967
- Setting up security on z/OS

If you are migrating from a previous version of IBM MQ for z/OS, you do not need to perform most of the customization tasks. See Maintaining and migrating for more information about the tasks you must perform.

Installable features of IBM MQ for z/OS

IBM MQ for z/OS comprises the following features:

Base
This is required; it comprises all the main functions, including

- Administration and utilities
- Support for CICS, IMS, and batch type applications using the IBM MQ Application Programming Interface, or C++
- Distributed queuing facility (supporting both TCP/IP and APPC communications)

National language features
These contain error messages and panels in all the supported national languages. Each language has a language letter associated with it. The languages and letters are:

C
Simplified Chinese

E
U.S. English (mixed case)

F
French

K
Japanese

U
U.S. English (uppercase)

You must install the US English (mixed case) option. You can also install one or more other languages. (The installation process for other languages requires US English (mixed case) to be installed, even if you are not going to use US English (mixed case).)

IBM MQ for z/OS UNIX System Services Components
This feature is optional. Select this feature if you want to build and run Java applications that use Jakarta Messaging 3.0 or Java Message Service 2.0 to connect to IBM MQ for z/OS.

See IBM MQ for z/OS Program Directory PDF files for information on installing IBM MQ for z/OS UNIX System Services Components.

IBM MQ for z/OS UNIX System Services Web Components
This feature is optional.

Select this feature if you want to use the IBM MQ Console, or the REST API.

You must install the IBM MQ for z/OS UNIX System Services Components feature, to install this feature.

IBM MQ for z/OS Managed File Transfer
This feature is optional, and should only be installed if you have entitlement for IBM MQ Advanced for z/OS, IBM MQ for z/OS Value Unit Edition (VUE), or IBM MQ for z/OS Managed File Transfer.

Select this feature if you want to use the Managed File Transfer capabilities of IBM MQ for z/OS.

You must install the IBM MQ for z/OS UNIX System Services Components feature, to install this feature.

Libraries that exist after installation

IBM MQ is supplied with a number of separate load libraries. Table 54 on page 898 shows the libraries that might exist after you have installed IBM MQ.

Table 54. IBM MQ libraries that exist after installation

Name	Description
thlqual.SCSQANLC	Contains the load modules for the Simplified Chinese version of IBM MQ.
thlqual.SCSQANLE	Contains the load modules for the U.S. English (mixed case) version of IBM MQ.
thlqual.SCSQANLF	Contains the load modules for the French version of IBM MQ.
thlqual.SCSQANLK	Contains the load modules for the Japanese version of IBM MQ.
thlqual.SCSQANLU	Contains the load modules for the U.S. English (uppercase) version of IBM MQ.
thlqual.SCSQASMS	Contains source for assembler sample programs.
thlqual.SCSQAUTH	The main repository for all IBM MQ product load modules; it also contains the default parameter module, CSQZPARM. This library must be APF-authorized and in PDS-E format.

Table 54. IBM MQ libraries that exist after installation (continued)	
Name	**Description**
thlqual.SCSQCICS	Contains extra load modules that must be included in the CICS DFHRPL concatenation. This library must be APF-authorized and in PDS-E format.
thlqual.SCSQCLST	Contains CLISTs used by the sample programs.
thlqual.SCSQCOBC	Contains COBOL copybooks, including copybooks required for the sample programs.
thlqual.SCSQCOBS	Contains source for COBOL sample programs.
thlqual.SCSQCPPS	Contains source for C++ sample programs.
thlqual.SCSQC37S	Contains source for C sample programs.
thlqual.SCSQC370	Contains C headers, including headers required for the sample programs.
thlqual.SCSQDEFS	Contains side definitions for C++ and the Db2 DBRMs for shared queuing.
thlqual.SCSQEXEC	Contains REXX executable files to be included in the SYSEXEC or SYSPROC concatenation if you are using the IBM MQ operations and control panels.
thlqual.SCSQFCMD	Contains templates for jobs to create and run Managed File Transfer tasks.
thlqual.SCSQHPPS	Contains header files for C++.
thlqual.SCSQINST	Contains JCL for installation jobs.
thlqual.SCSQLINK	Early code library. Contains the load modules that are loaded at system initial program load (IPL). The library must be APF-authorized.
thlqual.SCSQLOAD	Load library. Contains load modules for non-APF code, user exits, utilities, samples, installation verification programs, and adapter stubs. The library does not need to be APF-authorized and does not need to be in the link list. This library must be in PDS-E format.
thlqual.SCSQMACS	Contains Assembler macros including: sample macros, product macros, and system parameter macros.
thlqual.SCSQMAPS	Contains CICS mapsets used by sample programs.
thlqual.SCSQMSGC	Contains ISPF messages to be included in the ISPMLIB concatenation if you are using the Simplified Chinese language feature for the IBM MQ operations and control panels.
thlqual.SCSQMSGE	Contains ISPF messages to be included in the ISPMLIB concatenation if you are using the U.S. English (mixed case) language feature for the IBM MQ operations and control panels.
thlqual.SCSQMSGF	Contains ISPF messages to be included in the ISPMLIB concatenation if you are using the French language feature for the IBM MQ operations and control panels.
thlqual.SCSQMSGK	Contains ISPF messages to be included in the ISPMLIB concatenation if you are using the Japanese language feature for the IBM MQ operations and control panels.

Table 54. IBM MQ libraries that exist after installation (continued)

Name	Description
thlqual.SCSQMSGU	Contains ISPF messages to be included in the ISPMLIB concatenation if you are using the U.S. English (uppercase) language feature for the IBM MQ operations and control panels.
thlqual.SCSQMVR1	Contains the load modules for distributed queuing. This library must be APF-authorized and in PDS-E format.
thlqual.SCSQPLIC	Contains PL/I include files.
thlqual.SCSQPLIS	Contains source for PL/I sample programs.
thlqual.SCSQPNLA	Contains IPCS panels, for the dump formatter, to be included in the ISPPLIB concatenation. Also contains panels for IBM MQ sample programs.
thlqual.SCSQPNLC	Contains ISPF panels to be included in the ISPPLIB concatenation if you are using the Simplified Chinese language feature for the IBM MQ operations and control panels.
thlqual.SCSQPNLE	Contains ISPF panels to be included in the ISPPLIB concatenation if you are using the U.S. English (mixed case) language feature for the IBM MQ operations and control panels.
thlqual.SCSQPNLF	Contains ISPF panels to be included in the ISPPLIB concatenation if you are using the French language feature for the IBM MQ operations and control panels.
thlqual.SCSQPNLK	Contains ISPF panels to be included in the ISPPLIB concatenation if you are using the Japanese language feature for the IBM MQ operations and control panels.
thlqual.SCSQPNLU	Contains ISPF panels to be included in the ISPPLIB concatenation if you are using the U.S. English (uppercase) language feature for the IBM MQ operations and control panels.
thlqual.SCSQPROC	Contains sample JCL and default system initialization data sets.
thlqual.SCSQSNLC	Contains the load modules for the Simplified Chinese versions of the IBM MQ modules that are required for special purpose function (for example the early code).
thlqual.SCSQSNLE	Contains the load modules for the U.S. English (mixed case) versions of the IBM MQ modules that are required for special purpose function (for example the early code).
thlqual.SCSQSNLF	Contains the load modules for the French versions of the IBM MQ modules that are required for special purpose function (for example the early code).
thlqual.SCSQSNLK	Contains the load modules for the Japanese versions of the IBM MQ modules that are required for special purpose function (for example the early code).
thlqual.SCSQSNLU	Contains the load modules for the U.S. English (uppercase) versions of the IBM MQ modules that are required for special purpose function (for example the early code).
thlqual.SCSQTBLC	Contains ISPF tables to be included in the ISPTLIB concatenation if you are using the Simplified Chinese language feature for the IBM MQ operations and control panels.

Table 54. IBM MQ libraries that exist after installation (continued)	
Name	**Description**
thlqual.SCSQTBLE	Contains ISPF tables to be included in the ISPTLIB concatenation if you are using the U.S. English (mixed case) language feature for the IBM MQ operations and control panels.
thlqual.SCSQTBLF	Contains ISPF tables to be included in the ISPTLIB concatenation if you are using the French language feature for the IBM MQ operations and control panels.
thlqual.SCSQTBLK	Contains ISPF tables to be included in the ISPTLIB concatenation if you are using the Japanese language feature for the IBM MQ operations and control panels.
thlqual.SCSQTBLU	Contains ISPF tables to be included in the ISPTLIB concatenation if you are using the U.S. English (uppercase) language feature for the IBM MQ operations and control panels.

Note: Do not modify or customize any of these libraries. If you want to make changes, copy the libraries and make your changes to the copies.

Related concepts

IBM MQ for z/OS concepts

"Using IBM MQ with IMS" on page 1005
The IBM MQ -IMS adapter, and the IBM MQ - IMS bridge are the two components which allow IBM MQ to interact with IMS.

"Using IBM MQ with CICS" on page 1013
To use IBM MQ with CICS, you must configure the IBM MQ CICS adapter and, optionally, the IBM MQ CICS bridge components.

"Using OTMA exits in IMS" on page 1015
Use this topic if you want to use IMS Open Transaction Manager Access exits with IBM MQ for z/OS.

Related tasks

"Setting up communications with other queue managers on z/OS" on page 975
This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

Administering IBM MQ for z/OS

Related reference

"Upgrading and applying service to Language Environment or z/OS Callable Services" on page 1013
The actions you must take vary according to whether you use CALLLIBS or LINK, and your version of SMP/E.

z/OS Setting up IBM MQ for z/OS

Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

The best way to configure a queue manager is to carry out the following steps in the order shown:

1. Configure the base queue manager.
2. Configure the channel initiator, which performs queue manager to queue manager communications, and remote client application communication.
3. If you want to encrypt or protect messages, configure Advanced Message Security for z/OS.
4. If you want to use IBM MQ to transfer files, configure Managed File Transfer for z/OS.
5. If you want to use the administrative or messaging REST API, or the IBM MQ Console to manage IBM MQ from a web browser, configure the mqweb server.

This topic leads you through the various stages of setting up IBM MQ after you have successfully installed it. The installation process is described in the Program Directory. Para enlaces de descarga de los directorios de programas, consulteIBM MQ for z/OS Archivos PDF del directorio de programas .

Samples are supplied with IBM MQ to help you with your customization. The sample data set members have names beginning with the four characters CSQ4 and are in the library thlqual.SCSQPROC.

Before you perform the customization tasks described in this topic, there are a number of configuration options that you must consider because they affect the performance and resource requirements of IBM MQ for z/OS. For example, you must decide which globalization libraries you want to use.

If you want to automate some of the customization steps, see “Using IBM z/OSMF to automate IBM MQ ” on page 1019.

Configuration options

For more information about these options, see Planificación en z/OS.

The description of each task in this section indicates whether:

- The task is part of the process of setting up IBM MQ. That is, you perform the task once when you customize IBM MQ on the z/OS system. (In a parallel sysplex, you must perform the task for each z/OS system in the sysplex, and ensure that each z/OS system is set up identically.)
- The task is part of adding a queue manager. That is, you perform the task once for each queue manager when you add that queue manager.

None of the tasks require you to perform an IPL of your z/OS system, if you use commands to change the various z/OS system parameters, and perform “Update SYS1.PARMLIB members” on page 916 as suggested.

To simplify operations and to aid with problem determination, ensure that all z/OS systems in a sysplex are set up identically, so that queue managers can be quickly created on any system in an emergency.

For ease of maintenance, consider defining aliases to refer to your IBM MQ libraries; for more information, see Using an alias to refer to an IBM MQ library.

Related concepts

IBM MQ for z/OS concepts

“Using IBM MQ with IMS” on page 1005
The IBM MQ -IMS adapter, and the IBM MQ - IMS bridge are the two components which allow IBM MQ to interact with IMS.

“Using IBM MQ with CICS” on page 1013
To use IBM MQ with CICS, you must configure the IBM MQ CICS adapter and, optionally, the IBM MQ CICS bridge components.

“Using OTMA exits in IMS” on page 1015
Use this topic if you want to use IMS Open Transaction Manager Access exits with IBM MQ for z/OS.

Related tasks

“Setting up communications with other queue managers on z/OS” on page 975
This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

Administering IBM MQ for z/OS

Related reference

“Upgrading and applying service to Language Environment or z/OS Callable Services” on page 1013
The actions you must take vary according to whether you use CALLLIBS or LINK, and your version of SMP/E.

z/OS Configuring the z/OS system for IBM MQ

Use these topics as a step by step guide for customizing your IBM MQ for z/OS system.

z/OS *Identify the z/OS system parameters*

Some of the tasks involve updating the z/OS system parameters. You need to know which ones were specified when the system IPL was performed.

- *You need to perform this task once for each z/OS system where you want to run IBM MQ.*
- *You might need to perform this task when migrating from a previous version.*

SYS1.PARMLIB(IEASYSpp) contains a list of parameters that point to other members of SYS1.PARMLIB (where pp represents the z/OS system parameter list that was used to perform an IPL of the system).

The entries you need to find are:

For "Autorizar en APF las bibliotecas de carga de IBM MQ" on page 903:
PROG=xx or APF=aa point to the Authorized Program Facility (APF) authorized library list (member PROGxx or IEFAPFaa)

For "Actualizar LPA y la lista de enlaces de z/OS" on page 904:
LNK=kk points to the link list (member LNKLSTkk) LPA=mm points to the LPA list (member LPALSTmm)

For "Update the z/OS program properties table" on page 908:
SCH=xx points to the Program Properties Table (PPT) (member SCHEDxx)

For "Define the IBM MQ subsystem to z/OS" on page 909:
SSN=ss points to the defined subsystem list (member IEFSSNss)

z/OS *Autorizar en APF las bibliotecas de carga de IBM MQ*

APF autoriza varias bibliotecas. Es posible que algunos módulos de carga ya estén autorizados.

Notas:

- Debe realizar esta tarea una vez para cada sistema z/OS en el que desee ejecutar IBM MQ.
- Si está utilizando grupos de compartición de colas, debe asegurarse de que los valores para IBM MQ son idénticos en cada sistema z/OS del sysplex.
- Puede que sea necesario realizar esta tarea cuando se migra desde una versión anterior.
- Uso de LLA (Library Look aside):
  - Algún uso de IBM MQ puede hacer que la entrada/salida (E/S) alta cargue módulos de bibliotecas. Esta E/S se puede reducir utilizando el recurso LLA del sistema operativo.
  - Esta E/S alta puede producirse durante:
    - Aplicaciones con una tasa MQCONN/MQDISC alta, por ejemplo en un procedimiento almacenado WLM.
    - Cargando salidas de canal. Si tiene canales que se inician y detienen con frecuencia, y utiliza salidas de canal.
  - El miembro CSVLLAxx en SYS1.PARMLIB especifica la configuración de LLA. La inclusión de un nombre de biblioteca en la sentencia LIBRARIES significa que una copia de programa siempre se tomará de VLF (Virtual Lookaside Facility) y, por lo tanto, no necesitará E/S cuando se utilice mucho.

    La inclusión en la sentencia CONGELAR significa que no hay E/S para obtener los directorios de concatenación de sentencias DD relevantes (esto a menudo puede ser más E/S que la propia carga del programa).

    Utilice el mandato del sistema operativo "`F LLA,REFRESH`" después de realizar cualquier cambio en cualquiera de estas bibliotecas.

Las bibliotecas de carga de IBM MQ thlqual.SCSQAUTH y thlqual.SCSQLINK deben estar autorizadas por APF. También debe autorizar para APF las bibliotecas para la característica del idioma nacional (thlqual.SCSQANLx y thlqual.SCSQSNLx) y para la característica de gestión de colas distribuidas (thlqual.SCSQMVR1).

No obstante, todos los módulos de carga de LPA se autorizan automáticamente para APF. Al igual que todos los miembros de la lista de enlaces si el miembro SYS1.PARMLIB IEASYSpp contiene la sentencia:

```
LNKAUTH=LNKLST
```

LNKAUTH=LNKLST es el valor predeterminado si LNKAUTH no se especifica.

En función de lo que elija para poner en el LPA o en la lista de enlaces (consulte “Actualizar LPA y la lista de enlaces de z/OS” en la página 904), es posible que no sea necesario poner las bibliotecas de la lista de enlaces de APF

Nota: Debe autorizar para APF todas las bibliotecas que incluya en STEPLIB de IBM MQ. Si pone una biblioteca que no está autorizada por APF en STEPLIB, toda la concatenación de bibliotecas pierde su autorización APF.

Las listas de APF se encuentran en el sys1.parmlib miembro PROGxx o IEAAPFaa. Las listas contienen los nombres de las bibliotecas de z/OS autorizadas por APF. El orden de las entradas de las listas no es importante. Consulte Lista de bibliotecas autorizadas por APF para obtener información sobre las listas de APF.

Para obtener más información sobre cómo ajustar el sistema, consulte SupportPac MP16

Si utiliza miembros PROGxx con formato dinámico, sólo necesita emitir el mandato z/OS SETPROG APF,ADD,DSNAME=hlq.SCSQ *XXXX*,VOLUME= *YYYYYY* para que los cambios entren en vigor: donde *XXXX* varía según el nombre de biblioteca y donde *AAAA* es el volumen. De lo contrario, si utiliza el formato estático o miembros de IEAAPFaa, debe realizar una IPL en el sistema.

Tenga en cuenta que debe utilizar el nombre real de la biblioteca en la lista APF. Si intenta utilizar el alias del conjunto de datos de la biblioteca, la autorización fallará.

Conceptos relacionados

“Actualizar LPA y la lista de enlaces de z/OS” en la página 904
Actualice las bibliotecas de LPA con la nueva versión de bibliotecas de código inicial. Otro código pueden ir en la lista de enlaces o la LPA.

“Preparing to customize queue managers on z/OS” en la página 897
Use this topic when customizing your queue managers with details of installable features, national language features, and information about testing, and setting up security.

z/OS *Actualizar LPA y la lista de enlaces de z/OS*

Actualice las bibliotecas de LPA con la nueva versión de bibliotecas de código inicial. Otro código pueden ir en la lista de enlaces o la LPA.

- Debe realizar esta tarea una vez para cada sistema z/OS en el que desee ejecutar IBM MQ.
- Si está utilizando grupos de compartición de colas, debe renovar el código inicial en cada gestor de colas del QSG al nivel de IBM MQ 9.4.0 antes de migrar cualquiera de los gestores de colas a IBM MQ 9.4.0.

  Instale el código inicial más reciente en cada LPAR y renueve a continuación los gestores de colas, uno a uno, en algún momento antes de la migración. No es necesario migrar todos los gestores de colas al mismo tiempo.
- Puede que sea necesario realizar esta tarea cuando se migra desde una versión anterior. Para obtener más detalles, consulte el directorio del programa. Para enlaces de descarga de los directorios de programas, consulteIBM MQ for z/OS Archivos PDF del directorio de programas .

Nota: El conjunto de datos para LPA es específico de la versión. Si está utilizando un LPA existente en el sistema, póngase en contacto con el administrador del sistema para decidir qué LPA utilizar.

Código anterior

Algunos módulos de carga de IBM MQ se deben añadir a MVS para que IBM MQ actúe como un subsistema. Estos módulos se conocen con el nombre de Código inicial y se pueden ejecutar aunque un gestor de colas no esté activo. Por ejemplo, cuando un mandato de operador se emite en la consola con un prefijo de mandato de IBM MQ, este código inicial obtendrá el control y comprobará si tiene que iniciar un gestor de colas o bien pasar una solicitud para ejecutar un gestor de colas. Este código se carga en el Área de empaquetado de enlaces (LPA). Hay un conjunto de módulos iniciales, que se utilizan para todos los gestores de colas y tienen que estar en el último nivel de IBM MQ. El código inicial de una versión superior de IBM MQ funcionará con un gestor de colas que tenga una versión inferior de IBM MQ pero no funcionará con una versión superior.

El código inicial consta de los siguientes módulos de carga:

- CSQ3INI y CSQ3EPX en la biblioteca thqual.SCSQLINK
- CSQ3ECMX en la biblioteca thqual.SCSQSNL *x*, donde *x* es la letra del idioma.
  - thlqual.SCSQSNLE, para mayúsculas y minúsculas para el inglés de EE. UU.
  - thlqual.SCSQSNLU, para mayúsculas en inglés de EE.UU.
  - thlqual.SCSQSNLK, para japonés
  - thlqual.SCSQSNLF, para francés
  - thlqual.SCSQSNLC, para chino

IBM MQ incluye una modificación de usuario que mueve el contenido de la biblioteca thqual.SCSQSNL *i* a thqual.SCSQLINK e informa a SMP/E. Esta modificación de usuario se denomina CSQ8UERL y se describe en *Directorio del programa para IBM MQ for z/OS*, para Long Term Support o Continuous Delivery. Para enlaces de descarga de los directorios de programas, consulteIBM MQ for z/OS Archivos PDF del directorio de programas .

Una vez que haya actualizado el código inicial en las bibliotecas de LPA, estará disponible a partir de la siguiente IPL de z/OS (con la opción CLPA) para todos los subsistemas de gestor de colas añadidos durante la IPL en las definiciones de los miembros IEFSSNss en SYS1.PARMLIB.

Puede hacer que esté disponible inmediatamente sin una IPL para cualquier nuevo subsistema de gestor de colas añadido posteriormente (como se describe en “Define the IBM MQ subsystem to z/OS” en la página 909) añadiéndolo al LPA tal como se indica a continuación:

- Si no ha utilizado CSQ8UERL, emita estos mandatos de z/OS:

```
SETPROG LPA,ADD,MODNAME=(CSQ3INI,CSQ3EPX),DSNAME=thqual.SCSQLINK
SETPROG LPA,ADD,MODNAME=(CSQ3ECMX),DSNAME=thqual.SCSQSNL x
```

- Si ha utilizado CSQ8UERL, puede cargar el código inicial en el LPA utilizando el siguiente mandato de z/OS:

```
SETPROG LPA,ADD,MASK=*,DSNAME=thqual.SCSQLINK
```

- Si está utilizando Advanced Message Security también debe emitir el mandato z/OS siguiente para incluir un módulo adicional en el LPA:

```
SETPROG LPA,ADD,MODNAME=(CSQ0DRTM),DSNAME=thqual.SCSQLINK
```

Si ha aplicado mantenimiento, o tiene previsto reiniciar un gestor de colas con una versión o release posterior de IBM MQ, el código inicial puede estar disponible para los gestores de colas existentes utilizando los pasos siguientes. Los gestores de colas en los que no realiza estos pasos siguen utilizando la versión del código inicial que ya están utilizando. No es necesario realizar estos pasos para todos los gestores de colas en una LPAR, a menos que esté intentando específicamente aplicar mantenimiento a todos ellos, o actualizarlos todos a una versión o release más reciente de IBM MQ.

1. Añádalo al LPA mediante los mandatos SETPROG de z/OS tal como se describe anteriormente en este tema.
2. Detenga el gestor de colas con el mandato STOP QMGR de IBM MQ.
3. Asegúrese de que el perfil de seguridad qmgr.REFRESH.QMGR está configurado. Consulte Mandatos MQSC, perfiles y sus niveles de acceso.
4. Renueve el código inicial para el gestor de colas con el mandato REFRESH QMGR TYPE(EARLY) de IBM MQ.
5. Reinicie el gestor de colas con el mandato START QMGR de IBM MQ.

Los mandatos de IBM MQ STOP QMGR, REFRESH QMGR y START QMGR se describen en la sección Mandatos MQSC.

Otros códigos

Todos los módulos de carga suministrados por IBM MQ en las bibliotecas siguientes se han vuelto a introducir y se pueden colocar en el LPA:

- SCSQAUTH
- SCSQANL*x*, donde *x* es la letra de su idioma.
- SCSQMVR1

Importante: Sin embargo, si coloca las bibliotecas en el LPA, cada vez que aplique mantenimiento, tendrá que copiar manualmente los módulos modificados en el LPA. Por lo tanto, es preferible colocar las bibliotecas de carga de IBM MQ en la lista de enlaces, que se puede actualizar después del mantenimiento emitiendo el mandato z/OS MODIFY LLA REFRESH.

Consulte Modificación del contenido de conjuntos de datos LNKLST para obtener más información y Utilización del recurso LNKLST dinámico de forma segura y correcta.

Esto se recomienda especialmente para SCSQAUTH, ya que no tiene que incluirla en varias STEPLIB. Solo se debe colocar una biblioteca de idioma, SCSQANL*x* en el LPA o en la lista de enlaces. Las bibliotecas de lista de enlaces se especifican en un miembro LNKLSTkk de SYS1.PARMLIB.

El recurso de gestión de colas distribuidas y el CICS bridge (pero no el propio gestor de colas) necesitan acceso a la biblioteca de tiempo de ejecución de Language Environment (LE) SCEERUN. Si utiliza cualquiera de estos recursos, debe incluir SCEERUN en la lista de enlaces.

V 9.4.0 Algunos módulos se cargan al iniciar el gestor de colas en ECSA. En entornos restringidos ECSA, puede colocar estos módulos en el LPA en su lugar. Consulte “Placing IBM MQ global modules into the LPA” en la página 906.

Conceptos relacionados

“Update the z/OS program properties table” en la página 908
Some additional PPT entries are needed for the IBM MQ queue manager.

V 9.4.0 z/OS *Placing IBM MQ global modules into the LPA*

When an IBM MQ for z/OS queue manager starts up, it loads some of its load modules (global modules) into the extended common service area (ECSA). At queue manager shut down the ECSA is freed.

There are 19 global modules, which at IBM MQ 9.3, consumed approximately 1.2 MB of ECSA for each running queue manager.

Note: Although CSQ7GPLM is a global module, it should not be added to the LPA.

In environments that run multiple queue managers for each LPAR, and require a reduction in ECSA consumption due to ECSA or high private constraints, it is possible to place the global modules into the LPA. Placing the global modules of IBM MQ into the LPA is a manual process that requires care, so you should only carry out this procedure if there is a significant need to address ECSA or high private constraints.

If the queue manager cannot find a global module in its STEPLIB, and detects the module is in the LPA, it uses the LPA copy directly, instead of loading a copy of the module into ECSA. Alternatively if the queue managers code is normally loaded from the link list then any global modules in the LPA are loaded in preference to any global modules in the link list.

The z/OS common storage tracking function (see Using the common storage tracking function) tracks the storage under the each queue manager's MSTR address space for each queue manager and can be used to detect how much space is being use by the global modules.

By default, the global modules are in the SCSQAUTH load library. If the MSTR address space of a queue manager locates SCSQAUTH through the STEPLIB concatenation, the global modules from there are used in preference to any in the LPA and are loaded into ECSA.

The global modules are:

CSQ0GPLM, CSQ3AMGP, CSQ3SSGP, CSQ9PREP,
CSQ9SCNB, CSQGGPLM, CSQMCGLM, CSQMGPLM, CSQRGLM1,
CSQSLD1, CSQVGEPL, CSQVSRX, CSQWDLD2, CSQWDLD3,
CSQWVZSA, CSQWZDG0, CSQWVZPS, CSQWVGTM, CSQZTDDM

Important:

- The name of the global modules for IBM MQ remain constant across different IBM MQ versions. Therefore, if you load global modules into the LPA, they should be from a single IBM MQ version, and should only be used by queue managers running at the same IBM MQ version.
- If multiple versions of IBM MQ are run on the same LPAR then only one of those can have its global modules in the LPA at any given time.
- If maintenance is applied to an IBM MQ installation which has global modules loaded into the LPA, and that maintenance updates any of the global modules, you should perform the procedure described in the following text again.

Procedure

To put the global modules from a version of IBM MQ into the LPA, perform the following steps:

1. Create a copy of the `thlqual.SCSQAUTH` load library, and its contents, for example: `thlqual.LOCAL.SCSQAUTH`. Ensure that this load library is protected from unauthorized access using your external security manager (ESM).
2. APF authorize the `thlqual.LOCAL.SCSQAUTH` load library; see “Autorizar en APF las bibliotecas de carga de IBM MQ” on page 903.
3. Create a new `thlqual.GLOBAL.SCSQAUTH` load library with the same attributes as `thlqual.LOCAL.SCSQAUTH`.

   Note: This load library does not need to be APF authorized. Ensure that this load library is protected from unauthorized access using your ESM.
4. Copy the 19 global modules from `thlqual.LOCAL.SCSQAUTH` into `thlqual.GLOBAL.SCSQAUTH`.
5. Delete the 19 global modules from `thlqual.LOCAL.SCSQAUTH`.
6. Place the 19 global modules from `thlqual.GLOBAL.SCSQAUTH` into the LPA, by either:
   a. a. Adding `thlqual.GLOBAL.SCSQAUTH` into an `LPALSTxx` member of SYS1.PARMLIB. You must then IPL the system with the CLPA option to ensure that the library contents are loaded into the PLPA.
   b. b. Dynamically adding the modules to the LPA using the following command:

```
SETPROG
LPA,ADD,MODNAME=(CSQ0GPLM,CSQ3AMGP,CSQ3SSGP,CSQ9PREP,CSQ9SCNB,CSQGGPLM,
CSQMCGLM,CSQMGPLM,CSQRGLM1,CSQSLD1,CSQVGEPL,CSQVSRX,CSQWDLD2,CSQWDLD3,
CSQWVZSA,CSQWZDG0,CSQWVZPS,CSQWVGTM,CSQZTDDM),DSNAME= thlqual.GLOBAL.SCSQAUTH
```

Note: `LPALSTxx` is the preferred long term means of placing modules in LPA.

7. Validate that the modules are in the LPA by issuing the following command:

```
D PROG,LPA,MODNAME=CSQMCGLM
```

The output of the command should indicate the entry and load points of the module if it was successfully loaded into the LPA.

For each queue manager that needs to use the global modules from the LPA, then if you normally place:

1. `thlqual.SCSQAUTH` in the link list, just stop and start your queue manager. The global modules are loaded from the LPA, and the local modules from the link list.
2. `thlqual.SCSQAUTH` in the MSTR JCL STEPLIB, change the JCL so that the STEPLIB uses `thlqual.LOCAL.SCSQAUTH` instead of `thlqual.SCSQAUTH`. Stop and start the queue manager; the global modules are loaded from the LPA, and the local modules from the STEPLIB.

The CHIN and AMSM JCL can continue to use `thlqual.SCSQAUTH` as can any IBM MQ applications.

To revert the queue manager to loading the global modules into ECSA perform the following steps:

1. Stop the queue managers
2. Remove the global modules from the LPA, either at the next IPL by removing the LPALSTxx definitions or by using the following command:

```
SETPROG LPA,DELETE,MODNAME=(xxx) FORCE=YES
```

3. If `thlqual.LOCAL.SCSQAUTH` is in the STEPLIB of the queue manager replace it with `thlqual.SCSQAUTH`.
4. Restart the queue managers.

Related concepts

“Actualizar LPA y la lista de enlaces de z/OS” on page 904
Actualice las bibliotecas de LPA con la nueva versión de bibliotecas de código inicial. Otro código pueden ir en la lista de enlaces o la LPA.

z/OS *Update the z/OS program properties table*

Some additional PPT entries are needed for the IBM MQ queue manager.

- *You must perform this task once for each z/OS system where you want to run IBM MQ.*
- *If you are using queue sharing groups, you must ensure that the settings for IBM MQ are identical on each z/OS system in the sysplex.*
- *You do not need to perform this task when migrating from a previous version.*
- *You do need to perform the CSQ0DSRV part of this task when you require Advanced Message Security.*

A sample containing all the required PPT entries is provided in thlqual.SCSQPROC(CSQ4SCHD). Ensure that the required entries are added to the PPT, which you can find in SYS1.PARMLIB(SCHEDxx).

In z/OS, CSQYASCP is already defined to the operating system with the attributes detailed and no longer needs to be included in a SCHEDxx member of PARMLIB.

The IBM MQ queue manager controls swapping itself. However, if you have a heavily-loaded IBM MQ network and response time is critical, it might be advantageous to make the IBM MQ channel initiator nonswappable, by adding the CSQXJST PPT entry, at the risk of affecting the performance of the rest of your z/OS system.

If you require Advanced Message Security, add the CSQ0DSRV PPT entry.

Issue the z/OS command **SET SCH=xx**, where *xx* is the suffix of the SCHEDxx member of PARMLIB, for these changes to take effect.

Related concepts

“Define the IBM MQ subsystem to z/OS” on page 909
Update the subsystem name table and decide on a convention for command prefix strings.

z/OS Configuring the queue manager and channel initiator

Use these topics as a step by step guide for configuring the queue manager and channel initiator.

z/OS *Define the IBM MQ subsystem to z/OS*

Update the subsystem name table and decide on a convention for command prefix strings.

Repeat this task for each IBM MQ queue manager. You do not need to perform this task when migrating from a previous version.

Related concepts

"Create procedures for the IBM MQ queue manager" on page 912
Each IBM MQ subsystem needs a cataloged procedure to start the queue manager. You can create your own or use the IBM-supplied procedure library.

z/OS *Updating the subsystem name table*

When defining the IBM MQ subsystem you must add an entry to the subsystem name table.

The subsystem name table of z/OS, which is taken initially from the SYS1.PARMLIB member IEFSSNss, contains the definitions of formally defined z/OS subsystems. To define each IBM MQ subsystem, you must add an entry to this table, either by changing the IEFSSNss member of SYS1.PARMLIB, or, preferably, by using the z/OS command SETSSI.

IBM MQ subsystem initialization supports parallel processing, so IBM MQ subsystem definition statements can be added both above and below the BEGINPARALLEL keyword in the IEFSSNss table available at z/OS V1.12 and later.

If you use the SETSSI command, the change takes effect immediately, and there is no need to perform an IPL of your system. Ensure you update SYS1.PARMLIB as well, as described in "Update SYS1.PARMLIB members" on page 916 so that the changes remain in effect after subsequent IPLs.

The SETSSI command to dynamically define an IBM MQ subsystem is:

```
SETSSI ADD,S=ssid,I=CSQ3INI,P='CSQ3EPX,cpf,scope'
```

The corresponding information in IEFSSNss can be specified in one of two ways:

- The keyword parameter form of the IBM MQ subsystem definition in IEFSSNss. This is the recommended method.

```
SUBSYS SUBNAME(ssid) INITRTN(CSQ3INI) INITPARM('CSQ3EPX,cpf,scope')
```

- The positional parameter form of the IBM MQ subsystem definition.

```
ssid,CSQ3INI,'CSQ3EPX,cpf,scope'
```

Do not mix the two forms in one IEFSSNss member. If different forms are required, use a separate IEFSSNss member for each type, adding the SSN operand of the new member to the IEASYSpp SYS1.PARMLIB member. To specify more than one SSN, use SSN=(aa,bb,...) in IEASYSpp.

In the examples,

ssid
: The subsystem identifier. It can be up to four characters long. All characters must be alphanumeric (uppercase A through Z, 0 through 9), it must start with an alphabetic character. The queue manager

will have the same name as the subsystem, therefore you can use only characters that are allowed for both z/OS subsystem names and IBM MQ object names.

cpf
The command prefix string (see “Defining command prefix strings (CPFs)” on page 910 for information about CPFs).

scope
The system scope, used if you are running in a z/OS sysplex (see “CPFs in a sysplex environment” on page 911 for information about system scope).

Figure 97 on page 910 shows several examples of IEFSSNss statements.

```
CSQ1,CSQ3INI,'CSQ3EPX,+mqs1cpf,S'
CSQ2,CSQ3INI,'CSQ3EPX,+mqs2cpf,S'
CSQ3,CSQ3INI,'CSQ3EPX,++,S'
```

Figure 97. Sample IEFSSNss statements for defining subsystems

Note: When you have created objects in a subsystem, you cannot change the subsystem name or use the page sets from one subsystem in another subsystem. To do either of these, you must unload all the objects and messages from one subsystem and reload them into another.

Table 55 on page 910 gives a number of examples showing the associations of subsystem names and command prefix strings (CPFs), as defined by the statements in Figure 97 on page 910.

Table 55. Subsystem name to CPF associations

IBM MQ subsystem name	CPF
CSQ1	+mqs1cpf
CSQ2	+mqs2cpf
CSQ3	++

Note: The ACTIVATE and DEACTIVATE functions of the z/OS command SETSSI are not supported by IBM MQ.

To check the status of the changes, issue the following command in SDSF: `/D SSI,L`. You will see the new subsystems created with ACTIVE status.

z/OS *Defining command prefix strings (CPFs)*

Each subsystem instance of IBM MQ can have a command prefix string to identify that subsystem.

Adopt a system-wide convention for your CPFs for all subsystems to avoid conflicts. Adhere to the following guidelines:

- Define a CPF as string of up to eight characters.
- Do not use a CPF that is already in use by any other subsystem, and avoid using the JES backspace character defined on your system as the first character of your string.
- Define your CPF using characters from the set of valid characters listed in Table 57 on page 911.
- Do not use a CPF that is an abbreviation for an already defined process or that might be confused with command syntax. For example, a CPF such as 'D' conflicts with z/OS commands such as DISPLAY. To avoid this happening, use one of the special characters (shown in Table 57 on page 911) as the first or only character in your CPF string.
- Do not define a CPF that is either a subset or a superset of an existing CPF. For an example, see Table 56 on page 911.

Table 56. Example of CPF subset and superset rules		
Subsystem name	**CPF defined**	**Commands routed to**
MQA	!A	MQA
MQB	!B	MQB
MQC1	!C1	MQC1
MQC2	!C2	MQC2
MQB1	!B1	MQB

Commands intended for subsystem MQB1 (using CPF !B1) are routed to subsystem MQB because the CPF for this subsystem is !B, a subset of !B1. For example, if you entered the command:

```
!B1  START  QMGR
```

subsystem MQB receives the command:

```
1 START QMGR
```

(which, in this case, it cannot deal with).

You can see which prefixes exist by issuing the z/OS command DISPLAY OPDATA.

If you are running in a sysplex, z/OS diagnoses any conflicts of this type at the time of CPF registration (see "CPFs in a sysplex environment" on page 911 for information about CPF registration).

Table 57 on page 911 shows the characters that you can use when defining your CPF strings:

Table 57. Valid character set for CPF strings	
Character set	**Contents**
Alphabetic	Uppercase A through Z, lowercase a through z
Numeric	0 through 9
National (see note)	@ $ # (Characters that can be represented as hexadecimal values)
Special	. □ ( ) * & + - = ¢ < | ! ; % _ ? : >

Note:

The system recognizes the following hexadecimal representations of the national characters: @ as X'7C', $ as X'5B', and # as X'7B'. In countries other than the U.S., the U.S. national characters represented on terminal keyboards might generate a different hexadecimal representation and cause an error. For example, in some countries the $ character might generate an X'4A'.

The semicolon (;) is valid as a CPF but on most systems, this character is the command delimiter.

z/OS *CPFs in a sysplex environment*

Use this topic to understand how to use CPFs within the scope of a sysplex.

If used in a sysplex environment, IBM MQ registers your CPFs to enable you to enter a command from any console in the sysplex and route that command to the appropriate system for execution. The command responses are returned to the originating console.

Defining the scope for sysplex operation

Scope is used to determine the type of CPF registration performed by the IBM MQ subsystem when you are running IBM MQ in a sysplex environment.

Possible values for scope are as follows:

M

System scope.

The CPF is registered with z/OS at system IPL time by IBM MQ and remains registered for the entire time that the z/OS system is active.

IBM MQ commands must be entered at a console connected to the z/OS image running the target subsystem, or you must use ROUTE commands to direct the command to that image.

Use this option if you are not running in a sysplex.

S

Sysplex started scope.

The CPF is registered with z/OS when the IBM MQ subsystem is started, and remains active until the IBM MQ subsystem terminates.

You must use ROUTE commands to direct the original START QMGR command to the target system, but all further IBM MQ commands can be entered at any console connected to the sysplex, and are routed to the target system automatically.

After IBM MQ termination, you must use the ROUTE commands to direct subsequent START commands to the target IBM MQ subsystem.

X

Sysplex IPL scope.

The CPF is registered with z/OS at system IPL time by IBM MQ and remains registered for the entire time that the z/OS system is active.

IBM MQ commands can be entered at any console connected to the sysplex, and are routed to the image that is executing the target system automatically.

An IBM MQ subsystem with a CPF with scope of S can be defined on one or more z/OS images within a sysplex, so these images can share a single subsystem name table. However, you must ensure that the initial START command is issued on (or routed to) the z/OS image on which you want the IBM MQ subsystem to run. If you use this option, you can stop the IBM MQ subsystem and restart it on a different z/OS image within the sysplex without having to change the subsystem name table or perform an IPL of a z/OS system.

An IBM MQ subsystem with a CPF with scope of X can only be defined on one z/OS image within a sysplex. If you use this option, you must define a unique subsystem name table for each z/OS image requiring IBM MQ subsystems with CPFs of scope X.

If you want to use the z/OS automatic restart manager (ARM) to restart queue managers in different z/OS images automatically, every queue manager must be defined in each z/OS image on which that queue manager might be restarted. Every queue manager must be defined with a sysplex-wide, unique 4-character subsystem name with a CPF scope of S.

z/OS *Create procedures for the IBM MQ queue manager*

Each IBM MQ subsystem needs a cataloged procedure to start the queue manager. You can create your own or use the IBM-supplied procedure library.

- Repeat this task for each IBM MQ queue manager.
- You might need to modify the cataloged procedure when migrating from a previous version.

For each IBM MQ subsystem defined in the subsystem name table, create a cataloged procedure in a procedure library for starting the queue manager. The IBM-supplied procedure library is called SYS1.PROCLIB, but your installation might use its own naming convention.

The name of the queue manager started task procedure is formed by concatenating the subsystem name with the characters MSTR. For example, subsystem CSQ1 has the procedure name CSQ1MSTR. You need one procedure for each subsystem you define.

You need to include the library containing messages in your selected language:

- thlqual.SCSQSNLE, for US English mixed case
- thlqual.SCSQSNLU, for US English uppercase
- thlqual.SCSQSNLK, for Japanese
- thlqual.SCSQSNLF, for French
- thlqual.SCSQSNLC, for Chinese

Many examples and instructions in this product documentation assume that you have a subsystem called CSQ1. You might find these examples easier to use if a subsystem called CSQ1 is created initially for installation verification and testing purposes.

Two sample started task procedures are provided in thlqual.SCSQPROC. Member CSQ4MSTR uses one page set for each class of message, member CSQ4MSRR uses multiple page sets for the major classes of message. Copy one of these procedures to member xxxxMSTR (where xxxx is the name of your IBM MQ subsystem) of your SYS1.PROCLIB or, if you are not using SYS1.PROCLIB, your procedure library. Copy the sample procedure to a member in your procedure library for each IBM MQ subsystem that you define.

When you have copied the members, you can tailor them to the requirements of each subsystem, using the instructions in the member. For information about specifying limits of storage used by the queue manager, see Storage configuration. You can also use symbolic parameters in the JCL to allow the procedure to be modified when it is started. If you have several IBM MQ subsystems, you might find it advantageous to use JCL include groups for the common parts of the procedure, to simplify future maintenance.

If you are using queue sharing groups, the STEPLIB concatenation must include the Db2 runtime target library SDSNLOAD, and it must be APF-authorized. This library is only required in the STEPLIB concatenation if it is not accessible through the link list or LPA.

Notes:

1. You can make a note of the names of your bootstrap data set (BSDS), logs, and page sets for use in JCL and then define these sets at a later step in the process.
2. Sample started task procedures CSQ4MSTR and CSQ4MSRR have been updated to include, but leave commented out, the CSQMINI DD card that can be used to define a QMINI data set that contains transport security, that is, SSL or TLS properties.

   You can use “The QMINI data set” on page 919 to enable or disable TLS 1.3 support and/or be used to define a custom list of CipherSpecs to be used by channels.

Related concepts

“Create procedures for the channel initiator” on page 913
For each IBM MQ subsystem, tailor a copy of CSQ4CHIN. Depending on what other products you are using, you might need to allow access to other data sets.

z/OS *Create procedures for the channel initiator*

For each IBM MQ subsystem, tailor a copy of CSQ4CHIN. Depending on what other products you are using, you might need to allow access to other data sets.

- Repeat this task for each IBM MQ queue manager.
- You might need to modify the cataloged procedure when migrating from a previous version.

You need to create a channel-initiator started task procedure for each IBM MQ subsystem that is going to use distributed queuing.

To do this:

1. Copy the sample started task procedure thlqual.SCSQPROC(CSQ4CHIN) to your procedure library. Name the procedure *xxxx* CHIN, where *xxxx* is the name of your IBM MQ subsystem (for example, CSQ1CHIN would be the channel initiator started task procedure for queue manager CSQ1).
2. Make a copy for each IBM MQ subsystem that you are going to use.
3. Tailor the procedures to your requirements using the instructions in the sample procedure CSQ4CHIN. You can also use symbolic parameters in the JCL to allow the procedure to be modified when it is started. This is described with the start options in Administering IBM MQ for z/OS.

   Concatenate the distributed queuing library thlqual.SCSQMVR1.

   Access to the LE runtime library SCEERUN is required; if it is not in your link list (SYS1.PARMLIB(LNKLSTkk)), concatenate it in the STEPLIB DD statement.

   V 9.4.0 Consider adjusting the MEMLIMIT parameter using the information in Storage configuration.
4. Authorize the procedures to run under your external security manager.
5. You need to include the library containing messages in your selected language:
   - thlqual.SCSQSNLE, for US English mixed case
   - thlqual.SCSQSNLU, for US English uppercase
   - thlqual.SCSQSNLK, for Japanese
   - thlqual.SCSQSNLF, for French
   - thlqual.SCSQSNLC, for Chinese

The channel initiator is a long running address space. To prevent its termination after a restricted amount of CPU has been consumed, confirm that either:

- The default for started tasks in your z/OS system is unlimited CPU; a JES2 configuration statement for JOBCLASS(STC) with TIME=(1440,00) achieves this, or
- Explicitly add a TIME=1440, or TIME=NOLIMIT, parameter to the EXEC statement for CSQXJST.

You can add the exit library (CSQXLIB) to this procedure later if you want to use channel exits. You need to stop and restart your channel initiator to do this.

If you are using TLS, access to the system TLS runtime library is required. This library is called SIEALNKE. The library must be APF authorized.

If you are using TCP/IP, the channel initiator address space must be able to access the TCPIP.DATA data set that contains TCP/IP system parameters. The ways that the data set has to be set up depends on which TCP/IP product and interface you are using. They include:

- Environment variable, RESOLVER_CONFIG
- /etc/resolv.conf on the file system
- //SYSTCPD DD statement
- //SYSTCPDD DD statement
- *jobname/userid*.TCPIP.DATA
- SYS1.TCPPARMS(TCPDATA)
- *zapname*.TCPIP.DATA

Some of these affect your started-task procedure JCL. For more information, see z/OS Communications Server: IP Configuration Guide.

Related concepts

"Define the IBM MQ subsystem to a z/OS WLM service class" on page 915

To give IBM MQ appropriate performance priority in the z/OS system, you must assign the queue manager and channel initiator address spaces to an appropriate z/OS workload management (WLM) service class. If you do not do this explicitly, inappropriate defaults might apply.

z/OS *Define the IBM MQ subsystem to a z/OS WLM service class*

To give IBM MQ appropriate performance priority in the z/OS system, you must assign the queue manager and channel initiator address spaces to an appropriate z/OS workload management (WLM) service class. If you do not do this explicitly, inappropriate defaults might apply.

- *Repeat this task for each IBM MQ queue manager.*
- *You do not need to perform this task when migrating from a previous version.*

Use the ISPF dialog supplied with WLM to perform the following tasks:

- Extract the z/OS WLM policy definition from the WLM couple data set.
- Update this policy definition by adding queue manager and channel initiator started task procedure names to the chosen service class
- Install the changed policy on the WLM couple data set

Then activate this policy using the z/OS command

```
V WLM,POLICY=policyname,REFRESH
```

See Planning your IBM MQ environment on z/OS for more information on setting performance options.

Related concepts

“Set up the Db2 environment” on page 952
If you are using queue sharing groups you must create the required Db2 objects by customizing and running a number of sample jobs.

z/OS *Implement your ESM security controls*

Implement security controls for queue managers and the channel initiator.

- *Repeat this task for each IBM MQ queue manager.*
- *You might need to perform this task when migrating from a previous version.*

If you use RACF® as your external security manager, see Setting up security on z/OS , which describes how to implement these security controls.

If you are using the channel initiator, you must also do the following:

- If your subsystem has connection security active, define a connection security profile ssid.CHIN to your external security manager (see Connection security profiles for the channel initiator for information about this).
- If you are using Transport Layer Security (TLS) or a sockets interface, ensure that the user ID under whose authority the channel initiator is running is configured to use z/OS UNIX System Services, as described in the *z/OS UNIX System Services Planning* documentation.
- If you are using TLS, ensure that the user ID under whose authority the channel initiator is running is configured to access the key ring specified in the SSLKEYR parameter of the ALTER QMGR command.

Before you start the queue manager, set up IBM MQ data set and system security by:

- Authorizing the queue manager started task procedure to run under your external security manager.
- Authorizing access to the queue manager data sets.
- Configuring z/OS data set encryption if required.

  See the section, confidentiality for data at rest on IBM MQ for z/OS with data set encryption. for more information.

For details about how to do this, see Security installation tasks for z/OS.

If you are using RACF, provided you use the RACF STARTED class, you do not need to perform an IPL of your system (see RACF authorization of started-task procedures).

Related concepts

"Update SYS1.PARMLIB members" on page 916
To ensure that your changes remain in effect after an IPL, you must update some members of SYS1.PARMLIB

"Implement ESM security controls for the queue sharing group" on page 955
Implement security controls for all queue managers in a queue sharing group, to access Db2 and the coupling facility list structures.

z/OS *Update SYS1.PARMLIB members*

To ensure that your changes remain in effect after an IPL, you must update some members of SYS1.PARMLIB

- *You need to perform this task once for each z/OS system where you want to run IBM MQ.*
- *If you are using queue sharing groups, you must ensure that the settings for IBM MQ are identical on each z/OS system in the sysplex.*
- *You might need to perform this task when migrating from a previous version.*

Update SYS1.PARMLIB members as follows:

1. Update member IEFSSNss as described in "Define the IBM MQ subsystem to z/OS" on page 909.
2. Change IEASYSpp so that the following members are used when an IPL is performed:
   - the PROGxx or IEAAPFaa members used in "Autorizar en APF las bibliotecas de carga de IBM MQ" on page 903
   - the LNKLSTkk and LPALSTmm members used in "Actualizar LPA y la lista de enlaces de z/OS" on page 904
   - the SCHEDxx member used in "Update the z/OS program properties table" on page 908
   - the IEFSSNss member used in "Define the IBM MQ subsystem to z/OS" on page 909

Related concepts

"Customize the initialization input data sets" on page 916
Make working copies of the sample initialization input data sets and tailor them to suit your system requirements.

z/OS *Customize the initialization input data sets*

Make working copies of the sample initialization input data sets and tailor them to suit your system requirements.

- *Repeat this task for each IBM MQ queue manager.*
- *You need to perform this task when migrating from a previous version.*

Each IBM MQ queue manager gets its initial definitions from a series of commands contained in the IBM MQ *initialization input data sets*. These data sets are referenced by the DD names CSQINP1, CSQINP2, and CSQINPT defined in the queue manager started task procedure.

Responses to these commands are written to the initialization output data sets referenced by the DD names CSQOUT1, CSQOUT2 and CSQOUTT.

To preserve the originals, make working copies of each sample. Then you can tailor the commands in these working copies to suit your system requirements.

If you use more than one IBM MQ subsystem, if you include the subsystem name in the high-level qualifier of the initialization input data set name, you can identify the IBM MQ subsystem associated with each data set more easily.

Refer to the following topics for further information about the samples:

- Initialization data set formats
- Using the CSQINP1 sample
- Using the CSQINP2 samples
- Using the CSQINPX sample
- Using the CSQINPT sample

Initialization data set formats

The initialization input data sets can be partitioned data set (PDS) members or sequential data sets. They can be a concatenated series of data sets. Define them with a record length of 80 bytes, where:

- Only columns 1 through 72 are significant. Columns 73 through 80 are ignored.
- Records with an asterisk (*) in column 1 are interpreted as comments and are ignored.
- Blank records are ignored.
- Each command must start on a new record.
- A trailing - means continue from column 1 of the next record.
- A trailing + means continue from the first non-blank column of the next record.
- The maximum number of characters permitted in a command is 32 762.

The initialization output data sets are sequential data sets, with a record length of 125, a record format of VBA, and a block size of 629.

Using the CSQINP1 sample

Data set `thlqual.SCSQPROC` holds two members which contain definitions of buffer pools, page set to buffer pool associations, and an ALTER SECURITY command.

Member CSQ4INP1 uses one page set for each class of message. The messages are divided into the following classes:

- System-related messages.
- Important long-lived messages.
- Short-lived messages.
- Miscellaneous messages.

Member CSQ4INPR uses multiple page sets for each of the major classes of message, and one page set for each other class. The following are the major classes of messages:

- Important long-lived messages.
- Short-lived messages.

Include the appropriate sample in the CSQINP1 concatenation of your queue manager started task procedure.

Notes:

1. IBM MQ supports up to 100 buffer pools in the range zero through 99. The DEFINE BUFFPOOL command can only be issued from a CSQINP1 initialization data set. The definitions in the sample specify four buffer pools.
2. Each page set used by the queue manager must be defined in the CSQINP1 initialization data set by using the DEFINE PSID command. The page set definition associates a buffer pool ID with a page set. If no buffer pool is specified, buffer pool zero is used by default.

   Page set zero (00) must be defined. It contains all the object definitions. You can define up to 100 page sets for each queue manager.

3. The ALTER SECURITY command can be used to alter the security attributes TIMEOUT and INTERVAL. In CSQ4INP1, the default values are defined as 54 for TIMEOUT and 12 for INTERVAL.

See Planning your page sets and buffer pools for information about organizing buffer pools and page sets.

If you change the buffer pool and page set definitions dynamically while the queue manager is running, you should also update the CSQINP1 definitions. The changes are only retained for a cold start of IBM MQ, unless the buffer pool definition includes the REPLACE attribute.

Using the CSQINP2 samples

This table lists the members of `thlqual.SCSQPROC` that can be included in the CSQINP2 concatenation of your queue manager started task procedure, with a description of their function. The naming convention is CSQ4IN*. CSQ4INY* members should be modified for your configuration. You should avoid changing CSQINS* members because you will need to reapply any changes when you migrate to the next release. Instead, you can put DEFINE or ALTER commands in CSQ4INY* members.

Table 58. Members of `thlqual.SCSQPROC`

Member name	Description
CSQ4INSG	System object definitions.
CSQ4INSA	System object and default rules for channel authentication.
CSQ4INSX	System object definitions.
CSQ4INSS	Customize and include this member if you are using queue sharing groups.
CSQ4INSJ	Customize and include this member if you are using publish/subscribe using JMS.
CSQ4INSM	System object definitions for Advanced Message Security.
CSQ4INSR	Customize and include this member if you are using WebSphere Application Server, or the queued publish/subscribe interface supported by the queued publish/subscribe daemon in IBM MQ.
CSQ4DISP	CSQINP2 sample for displaying object definitions.
CSQ4INYC	Clustering definitions.
CSQ4INYD	Distributed queuing definitions.
CSQ4INYG	General definitions.
CSQ4INYR	Storage class definitions, using multiple page sets for the major classes of message.
CSQ4INYS	Storage class definitions, using one page set for each class of message.

You need to define objects once only, not each time that you start a queue manager, so it is not necessary to include these definitions in CSQINP2 every time. If you do include them every time, you are attempting to define objects that already exist, and you will get messages similar to the following:

```
CSQM095I +CSQ1 CSQMAQLC QLOCAL(SYSTEM.DEFAULT.LOCAL.QUEUE) ALREADY EXISTS
CSQM090E +CSQ1 CSQMAQLC FAILURE REASON CODE X'00D44003'
CSQ9023E +CSQ1 CSQMAQLC ' DEFINE QLOCAL' ABNORMAL COMPLETION
```

The objects are not damaged by this failure. If you want to leave the SYSTEM definitions data set in the CSQINP2 concatenation, you can avoid the failure messages by specifying the REPLACE attribute against each object.

Using the CSQINPX sample

Sample `thlqual.SCSQPROC(CSQ4INPX)` contains a set of commands that you might want to execute each time the channel initiator starts. These are typically channel-related commands such as START

LISTENER, which are required every time the channel initiator starts, rather than whenever the queue manager starts, and which are not allowed in the input data sets CSQINP1 or CSQINP2. You must customize this sample before use; you can then include it in the CSQINPX data set for the channel initiator.

The IBM MQ commands contained in the data set are executed at the end of channel initiator initialization, and output is written to the data set specified by the CSQOUTX DD statement. The output is like that produced by the COMMAND function of the IBM MQ utility program (CSQUTIL). See Using the CSQUTIL utility for IBM MQ for z/OS.

You can specify any of the IBM MQ commands that can be issued from CSQUTIL, not only the channel commands. You can enter commands from other sources while CSQINPX is being processed. All commands are issued in sequence, regardless of the success of the previous command.

To specify a command response time, you can use the pseudo-command COMMAND as the first command in the data set. This takes a single optional keyword RESPTIME(*nnn*), where *nnn* is the time, in seconds, to wait for a response to each command. This is in the range 5 through 999; the default is 30.

If IBM MQ detects that the responses to four commands have taken too long, processing of CSQINPX is stopped and no further commands are issued. The channel initiator is not stopped, but message CSQU052E is written to the CSQOUTX data set, and message CSQU013E is sent to the console.

When IBM MQ has completed processing of CSQINPX successfully, message CSQU012I is sent to the console.

Using the CSQINPT sample

This table lists the members of `thlqual.SCSQPROC` that can be included in the CSQINPT concatenation of your queue manager started task procedure, with a description of their function.

Table 59. Members of `thlqual.SCSQPROC`

Member name	Description
CSQ4INST	System default subscription definition.
CSQ4INYT	Publish/Subscribe definitions.

The IBM MQ commands contained in the data set are executed when publish/subscribe initialization completes, and output is written to the data set specified by the CSQOUTT DD statement. The output is like that produced by the COMMAND function of the IBM MQ utility program (CSQUTIL). See Using the CSQUTIL utility for IBM MQ for z/OS.

Related concepts

“Create the bootstrap and log data sets” on page 921
Use the supplied program CSQJU003 to prepare the bootstrap data sets (BSDSs) and log data sets.

z/OS *The QMINI data set*

You can use the QMINI data set to specify properties that are to be read and processed during queue manager initialization.

Characteristics of the QMINI data set

The QMINI data set is a sequential data set, or a member of a partitioned data set, with a maximum record length of 80 bytes (72 bytes for data and eight bytes for the line number).

The following example shows the properties for a sequential QMINI data set. Some properties are, of course, based on your environment.

```
Data Set Name . . . . : QM01.QMINI
General Data                          Current Allocation
Management class . .  : STANDARD      Allocated tracks  . : 1
Storage class  . . .  : STANDARD      Allocated extents . : 1
```

```
Volume serial . . .   : P5P21E
Device type . . . .   : 3390
Data class . . . . .  : **None**
Organization  . . .   : PS          Current Utilization
Record format . . .   : FB          Used tracks . . . . : 0
Record length . . .   : 80          Used extents  . . . : 0
Block size  . . . .   : 3120
1st extent tracks .   : 1
Secondary tracks  .   : 1           Dates
Data set name type    :             Creation date . . . : 2020/08/11
Data set encryption   : NO          Referenced date . . : ***None***
                                    Expiration date . . : ***None***
SMS Compressible  .   : NO
```

`thlqual.SCSQPROC`, includes:

- The sample contents for a QMINI data set in CSQ4QMIN.
- An example of specifying the QMINI data set using the //CSQMINI DD card, in the queue manager startup JCL, in the started task procedures CSQ4MSTR and CSQ4MSRR.

Notes:

- The code that parses the data set only parses the first 72 bytes of each record.
- Line numbers are ignored so it is not necessary to specify line numbers.
- If a line starts with an asterisk character (*), the line is treated as a comment.
- The contents of the QMINI data set are parsed during queue manager startup. If the contents are parsed successfully, message CSQM578I is issued in the queue manager job log. If any errors are encountered during parsing, error messages, for example CSQM573E, are issued in the queue manager job log but the queue manager still starts.

  Check for error messages, and resolve any issues in the contents of the QMINI data set.

  If the queue manager is unable to parse the QMINI data set, you can start the channel initiator, but you cannot start any channels that are configured to use SSL or TLS as the security configuration settings are unknown.
- If you make any updates to the data set after you have started the queue manager, you must restart the queue manager to pick up the changes.

The TransportSecurity stanza

From IBM MQ for z/OS 9.2.0, the QMINI data set supports the `TransportSecurity` stanza. This stanza provides similar function to that provided by the SSL stanza in the qm.ini file on IBM MQ for Multiplatforms.

The `TransportSecurity` stanza supports the following properties:

AllowTLSV13
Whether a queue manager is able to use the TLS 1.3 CipherSpecs; valid values are: *TRUE/T/YES/Y* or *FALSE/F/NO/N*.

For migrated queue managers, TLS 1.3 is not enabled by default. You can enable TLS 1.3 by defining a QMINI data set with the `TransportSecurity` stanza and **`AllowTLSV13`**=TRUE.

For newly created queue managers TLS 1.3 is enabled by default.

AllowedCipherSpecs
A custom list of CipherSpecs that are enabled.

See Providing a custom list of ordered and enabled CipherSpecs on IBM MQ for z/OS for more information on this property.

Duplicate CipherSpec names in the list are ignored.

OutboundSNI
Whether the Server Name Indication (SNI) is set to the target IBM MQ channel name to the remote system when initiating a TLS connection, or to the hostname; valid values are: `CHANNEL` or `HOSTNAME`.

If the destination channel is configured with a certificate label on the channel object CERTLABL field, you must set CERTLABL to the channel value. If a connection with a setting of HOSTNAME is made to a channel with a CERTLABL setting, the connection fails and an AMQ9673 message is printed in the remote queue manager error logs.

The following example shows how the `TransportSecurity` stanza is specified:

```
TransportSecurity:
AllowedCipherSpecs=ECDHE_RSA_AES_128_GCM_SHA256,
                   ECDHE_RSA_AES_256_GCM_SHA384
AllowTLSV13=TRUE
```

z/OS *Create the bootstrap and log data sets*

Use the supplied program CSQJU003 to prepare the bootstrap data sets (BSDSs) and log data sets.

Note:

- Repeat this task for each IBM MQ queue manager.
- If you are using z/OS data set encryption to protect the BSDS or active log data sets, you must configure this option before the data sets are allocated in this step.
- You do not need to perform this task when migrating from a previous version.
- If you are migrating a queue manager and adding z/OS data set encryption for active log data sets or BSDS, you need to convert the data sets.
- For more information about configuring z/OS data set encryption, and converting existing IBM MQ data sets to be encrypted, see confidentiality for data at rest on IBM MQ for z/OS with data set encryption..

The sample JCL and Access Method Services (AMS) control statements to run CSQJU003 to create a single or dual logging environment are held in thlqual.SCSQPROC(CSQ4BSDS). Customize and run this job to create your BSDSs and logs and to preformat the logs.

Important: You should use the newest version of CSQ4BSDS, or update your JCL manually to use RECORDS(850 60).

The started task procedure, CSQ4MSTR, described in “Create procedures for the IBM MQ queue manager” on page 912, refers to BSDSs in statements of the form:

```
//BSDS1   DD DSN=++HLQ++.BSDS01,DISP=SHR
//BSDS2   DD DSN=++HLQ++.BSDS02,DISP=SHR
```

The log data sets are referred to by the BSDSs.

Note:

1. The BLKSIZE must be specified on the SYSPRINT DD statement in the LOGDEF step. The BLKSIZE must be 629.
2. To help identify bootstrap data sets and log data sets from different queue managers, include the subsystem name in the high level qualifier of these data sets.
3. If you are using queue sharing groups, you must define the bootstrap and log data sets with SHAREOPTIONS(2 3).

See Planificación en z/OS for information about planning bootstrap and log data sets and their sizes.

From IBM MQ 8.0, the 8 byte log RBA enhancement improves the availability of a queue manager, as described in Larger log Relative Byte Address. To enable 8 byte log RBA on a queue manager before the queue manager is first started, perform the following steps after creating your logging environment.

Note: For queue managers created at IBM MQ 9.3.0 or later 8 byte log RBA is already enabled, so the following steps are not necessary.

1. Using **IDCAMS ALTER**, rename the version 1 format BSDSs (created using the CSQJU003 program) to something like ++HLQ++.V1.BSDS01.

   Note: Ensure that you rename the data and index components as well as the VSAM cluster.
2. Allocate new BSDSs with the same attributes as the ones already defined. These will become the version 2 format BSDSs that will be used by the queue manager when it is started.
3. Run the BSDS conversion utility (CSQJUCNV) to convert the version 1 format BSDSs to the new version 2 format BSDSs.
4. Once the conversion completes successfully, delete the version 1 format BSDSs.

Note: If the queue manager is in a queue sharing group, all queue managers in the queue sharing group must have been started as follows before 8 byte log RBA can be enabled:

- If the queue manager is at IBM MQ 9.0.0 LTS it must have been started with **OPMODE**(NEWFUNC,900) or **OPMODE**(*NEWFUNC,800*)
- If the queue manager is at IBM MQ 9.0.n CD, or IBM MQ 9.1.0 LTS, or later, it needs to have been started at that level

Related concepts

"Define your page sets" on page 922
Define page sets for each queue manager using one of the supplied samples.

z/OS *Define your page sets*

Define page sets for each queue manager using one of the supplied samples.

- *Repeat this task for each IBM MQ queue manager.*

  If you using z/OS data set encryption to protect the page sets, you must configure this option before the data sets are allocated in this step.
- *You do not need to perform this task when migrating from a previous version.*

  If you are migrating a queue manager and adding z/OS data set encryption for page sets, you need to convert the page sets.

  See the section, Confidentiality for data at rest on IBM MQ for z/OS with data set encryption. *for more information about configuring z/OS data set encryption and converting existing IBM MQ data sets to be encrypted.*

Define separate page sets for each IBM MQ queue manager. thlqual.SCSQPROC(CSQ4PAGE) and thlqual.SCSQPROC(CSQ4PAGR) contain JCL and z/OS access method services (AMS) control statements to define and format page sets. Member CSQ4PAGE uses one page set for each class of message, member CSQ4PAGR uses multiple page sets for the major classes of message. The JCL runs the supplied utility program CSQUTIL. Review the samples and customize them for the number of page sets you want and the sizes to use. See Planning your page sets and buffer pools for information about page sets and how to calculate suitable sizes.

The started task procedure CSQ4MSTR described in "Create procedures for the IBM MQ queue manager" on page 912 refers to the page sets, in a statement of the form:

```
//CSQP00nn DD DISP=OLD,DSN=xxxxxxxxxx
```

where *nn* is the page set number between 00 and 99, and *xxxxxxxxxx* is the data set that you define.

Note:

1. If you intend to use the dynamic page set expansion feature, ensure that secondary extents are defined for each page set. thlqual.SCSQPROC(CSQ4PAGE) shows how to do this.
2. To help identify page sets from different queue managers, include the subsystem name in the high level qualifier of the data set associated with each page set.

3. If you intend to allow the FORCE option to be used with the FORMAT function of the utility program CSQUTIL, you must add the REUSE attribute on the AMS DEFINE CLUSTER statement.

   See the Optional Parameters section of the z/OS DEFINE CLUSTER command for more information on REUSE.

4. If your page sets are to be larger than 4 GB you must use the Storage Management System (SMS) EXTENDED ADDRESSABILITY function.

Related concepts

"Add the IBM MQ entries to the Db2 tables" on page 955
If you are using queue sharing groups, run the CSQ5PQSG utility to add queue sharing group and queue manager entries to the IBM MQ tables in the Db2 data-sharing group.

z/OS *Tailor your system parameter module*

The IBM MQ system parameter module controls the logging, archiving, tracing, and connection environments that IBM MQ uses in its operation. A default module is supplied. You should create your own system parameter module as some parameters, for example data set names, are usually site specific.

- *Repeat this task for each IBM MQ queue manager, as required.*
- *You might need to perform this task when migrating from a previous version. For details, see* Migrating IBM MQ on z/OS.
- *To enable Advanced Message Security for z/OS on an existing queue manager, you only need to set SPLCAP to YES as described in* "Utilización de CSQ6SYSP" on page 925*. If you are configuring this queue manager for the first time, complete the whole of this task.*

The system parameter module has four macros as follows:

Macro name	Purpose
CSQ6SYSP	Specifies the connection and tracing parameters, see "Utilización de CSQ6SYSP" on page 925
CSQ6LOGP	Controls log initialization, see "Using CSQ6LOGP" on page 935
CSQ6ARVP	Controls archive initialization, see "Using CSQ6ARVP" on page 939
CSQ6USGP	Controls usage recording, see "Using CSQ6USGP" on page 946

IBM MQ supplies a default system parameter module, CSQZPARM, which is invoked automatically if you issue the START QMGR command (without a PARM parameter) to start an instance of IBM MQ. CSQZPARM is in the APF-authorized library thlqual.SCSQAUTH also supplied with IBM MQ. The values of these parameters are displayed as a series of messages when you start IBM MQ.

See START QMGR for more information about how this command is used.

Creating your own system parameter module

If CSQZPARM does not contain the system parameters you want, you can create your own system parameter module using the sample JCL provided in thlqual.SCSQPROC(CSQ4ZPRM).

To create your own system parameter module:

1. Make a working copy of the JCL sample.
2. Edit the parameters for each macro in the copy as required. If you remove any parameters from the macro calls, the default values are automatically picked up at run time.
3. Replace the placeholder ++NAME++ with the name that the load module is to take (this can be CSQZPARM).
4. If your assembler is not high-level assembler, change the JCL as required by your assembler.

5. Run the JCL to assemble and link edit the tailored versions of the system parameter macros to produce a load module. This is the new system parameter module with the name that you have specified.
6. Put the load module produced in an APF-authorized user library.
7. Add user READ access to the APF-authorized user library.
8. Include this library in the IBM MQ queue manager started task procedure STEPLIB. This library name must come before the library thlqual.SCSQAUTH in STEPLIB.
9. Invoke the new system parameter module when you start the queue manager. For example, if the new module is named NEWMODS, issue the command:

```
START QMGR PARM(NEWMODS)
```

10. Ensure successful completion of the command by checking the job log. There should be an entry in the log similar to the following:

```
CSQ9022I CDL1 CSQYASCP 'START QMGR' NORMAL COMPLETION
```

You can also specify the parameter module name in the queue manager startup JCL. For more information, see Using MQSC to start and stop a queue manager on z/OS.

Note: If you choose to name your module CSQZPARM, you do not need to specify the PARM parameter on the START QMGR command.

Fine tuning a system parameter module

IBM MQ also supplies a set of three assembler source modules, which can be used to fine-tune an existing system parameter module. These modules are in library thlqual.SCSQASMS. Typically, you use these modules in a test environment to change the default parameters in the system parameter macros. Each source module calls a different system parameter macro:

This assembler source module...	Calls this macro...
CSQFSYSP	CSQ6SYSP (connection and tracing parameters)
CSQJLOGP	CSQ6LOGP (log initialization)
CSQJARVP	CSQ6ARVP (archive initialization)

This is how you use these modules:

1. Make working copies of each assembler source module in a user assembler library.
2. Edit your copies by adding or altering the values of any parameters as required.
3. Assemble your copies of any edited modules to create object modules in a user object library.
4. Link edit these object code modules with an existing system parameter module to produce a load module that is the new system parameter module.
5. Ensure that new system parameter module is a member of a user authorized library.
6. Include this library in the queue manager started task procedure STEPLIB. This library must come before the library thlqual.SCSQAUTH in STEPLIB.
7. Invoke the new system parameter module by issuing a START QMGR command, specifying the new module name in the PARM parameter, as before.

A sample usermod is provided in member CSQ4UZPR of SCSQPROC which demonstrates how to manage customized system parameters under SMP/E control.

Altering system parameters

You can alter some system parameters while a queue manager is running; see the SET SYSTEM, SET LOG, and SET ARCHIVE commands.

Put the SET commands in your initialization input data sets so that they take effect every time you start the queue manager.

Related concepts

“Tailor the channel initiator parameters” on page 947
Use ALTER QMGR to customize the channel initiator to suit your requirements.

z/OS *Utilización de CSQ6SYSP*

Utilice este tema como referencia para conocer cómo establecer parámetros del sistema utilizando CSQ6SYSP.

Los parámetros predeterminados de CSQ6SYSP, y la indicación de si puede alterar cada parámetro mediante el mandato SET SYSTEM, se muestran en la Tabla 60 en la página 925. Si desea cambiar cualquiera de estos valores, vea las descripciones detalladas de los parámetros.

Tabla 60. Valores predeterminados de parámetros de CSQ6SYSP

Parámetro	Descripción	Valor predeterminado	Mandato SET
“ACCTIME” en la página 927	La hora, en minutos y segundos, entre cada recopilación de datos contables.	-1	✓
“ACELIM” en la página 927	Tamaño de la agrupación de almacenamiento ACE en bloques de 1 KB.	0 (sin límite)	✓
“CLCACHE” en la página 928	Especifica el tipo de memoria caché de clúster que se debe utilizar.	STATIC	-
“CMDUSER” en la página 928	ID de usuario predeterminado para las comprobaciones de seguridad de mandato.	CSQOPR	-
“EXCLMSG” en la página 928	Especifica una lista de mensajes que se han de excluir de cualquier registro. Los mensajes de esta lista no se envían al registro de copia impresa ni a la consola de z/OS. Por tanto, la utilización del parámetro EXCLMSG para excluir mensajes resulta más eficaz desde el punto de vista de la CPU que utilizar los métodos descritos en “Suppress information messages” en la página 951.	( )	✓
“EXITLIM” en la página 929	Tiempo máximo (en segundos) durante el que se pueden ejecutar las salidas del gestor de colas durante cada invocación.	30	-
“EXITTCB” en la página 929	Número de tareas de servidor iniciadas que se deben utilizar para ejecutar los programas de salida del gestor de colas.	8	-
“LOGLOAD” en la página 929	Número de registros de anotaciones escritos por IBM MQ entre el inicio de un punto de comprobación y el siguiente.	500 000	✓
“MULCCAPT” en la página 930	Controla la propiedad Measured Usage Pricing, que establece el algoritmo para recoger los datos utilizados por Measured Usage License Charging (MULC).	Consulte la descripción de parámetro	-

Tabla 60. Valores predeterminados de parámetros de CSQ6SYSP (continuación)			
Parámetro	**Descripción**	**Valor predeterminado**	**Mandato SET**
"OTMACON" en la página 930	Parámetros de conexión de OTMA.	Consulte la descripción de parámetro	-
"QINDXBLD" en la página 931	Determina si el reinicio del gestor de colas espera a que estén reconstruidos todos los índices, o el reinicio se completa antes de que estén reconstruidos todos los índices.	WAIT	-
"QMCCSID" en la página 931	Identificador de conjunto de caracteres codificados del gestor de colas.	Cero	-
"QSGDATA" en la página 931	Parámetros del grupo de compartición de colas.	Consulte la descripción de parámetro	-
"RESAUDIT" en la página 932	Parámetro de auditoría RESLEVEL.	SÍ	-
"ROUTCDE" en la página 932	Código de direccionamiento de mensajes asignado a los mensajes no solicitados desde una consola específica.	1	-
"SERVICIO" en la página 932	Reservado para uso de IBM.	0	✓
"SMFACCT" en la página 932	Especifica si SMF debe recoger datos contables cuando se inicia el gestor de colas. Tenga en cuenta que los datos de contabilidad del canal de clase 4 solo se recopilan cuando se inicia el iniciador de canal.	NO	-
SMFSTAT	Especifica si SMF debe recoger datos estadísticos cuando se inicia el gestor de colas. Tenga en cuenta que los datos de estadísticas del canal de clase 4 solo se recopilan cuando se inicia el iniciador de canal.	NO	-
SPLCAP	Especifica si la función de política de seguridad de cola está habilitada en este gestor de colas. En Advanced Message Security for z/OS, establezca este parámetro a YES.	NO	-
STATIME	El tiempo, en minutos y segundos, entre cada recopilación de estadísticas.	30	✓
TRACSTR	Especifica si se debe iniciar automáticamente el rastreo.	NO	-
TRACTBL	Tamaño de la tabla de rastreo, en bloques de 4 KB, que debe ser utilizada por el recurso de rastreo global.	99 (396 KB)	✓
WLMTIME	Tiempo transcurrido entre cada exploración del índice de cola para colas gestionadas por WLM.	30	-

Tabla 60. Valores predeterminados de parámetros de CSQ6SYSP (continuación)			
Parámetro	**Descripción**	**Valor predeterminado**	**Mandato SET**
WLMTIMU	Unidades (minutos o segundos) utilizadas para WLMTIME.	MINS	-

ACCTIME

Especifica el intervalo, en minutos y segundos, entre recopilaciones consecutivas de datos contables.

Especifique un número, `-1`, o en el rango de `0` a 1440 minutos en el formato `'mmmm'`, o en el rango de `0` a 1440 minutos, y de `0` a 59 segundos, en el formato `'mmmm.ss'`.

Notas:

- Cuando solo se especifica un intervalo de segundos, se debe añadir al intervalo un valor de `0` como prefijo. El intervalo más pequeño posible es un segundo: "`0.01`".
- Si especifica un valor de `0`, los datos contables se recopilan en el intervalo de registro global SMF. Consulte Utilización de System Management Facility si desea más información.
- Si especifica un valor de `-1`, que es el valor predeterminado, los datos contables se recopilan en el intervalo especificado por el valor STATIME.

Por ejemplo:

`'0.30'` establece un intervalo de 30 segundos.
`'5.30'` establece un intervalo de 5 minutos y 30 segundos.
`'30'` establece un intervalo de 30 minutos.

ACELIM

Especifica el tamaño máximo de la agrupación de almacenamiento ACE en bloques de 1 KB. El número debe estar en el intervalo de 0 a 999999. El valor predeterminado de cero significa ninguna restricción impuesta, más allá de lo que está disponible en el sistema.

Solo debe establecer un valor para ACELIM en los gestores de colas que se han identificado utilizando cantidades exorbitantes de almacenamiento ECSA. Limitar la agrupación de almacenamiento ACE tiene el efecto de limitar el número de conexiones del sistema y, por lo tanto, la cantidad de almacenamiento ECSA utilizado por un gestor de colas.

Cuando el gestor de colas alcanza el límite, las aplicaciones no pueden obtener nuevas conexiones. La falta de nuevas conexiones crea errores en el proceso MQCONN y las aplicaciones que se coordinan mediante RRS pueden sufrir anomalías en cualquier API de IBM MQ.

Una ACE representa aproximadamente el 12,5% del total de ECSA necesario para los bloques de control relacionados con la hebra para una conexión. Por lo tanto, por ejemplo, si se especifica ACELIM=5120 se esperaría que se limitara la cantidad total de ECSA asignada por el gestor de colas (para bloques de control relacionados con hebras) a aproximadamente 40960K; es decir, 5120 multiplicado por 8.

Para limitar la cantidad total de ECSA asignada por el gestor de colas, para los bloques de control relacionados con hebras en 5120K, se necesita un valor ACELIM de 640.

Puede utilizar registros SMF 115 subtipo 5, generados por rastreo de estadísticas CLASS(3), para supervisar el tamaño de la agrupación de almacenamiento 'ACE/PEB' y, por tanto, establecer un valor adecuado para ACELIM.

Puede obtener la cantidad total de almacenamiento ECSA utilizado por el gestor de colas, para bloques de control, de registros SMF 115 de subtipo 7, escritos por el rastreo de estadísticas CLASS(2). La cantidad total de almacenamiento ECSA utilizado es la suma de los campos QSRSPHBGF y QSRSPHBGV.

Para obtener más información sobre los registros de estadísticas de SMF 115, consulte Interpretación de estadísticas del rendimiento de IBM MQ.

Tenga en cuenta que debe considerar el establecer ACELIM como un mecanismo para proteger una imagen de z/OS ante un comportamiento incorrecto de un gestor de colas, en lugar de como un medio de controlar las conexiones de aplicación con un gestor de colas.

CLCACHE
Especifica el tipo de memoria caché de clúster que se debe utilizar.

La memoria caché de clúster es un área de almacenamiento utilizada para almacenar información relacionada con el clúster.

Si la memoria caché de clúster es estática, tiene un tamaño fijo, que se asigna al iniciar el gestor de colas. Si la memoria caché se llena, se emite el mensaje CSQM060E y la solicitud de aplicación que requería más espacio recibe un MQRC_CLUSTER_RESOURCE_ERROR.

Si establece CLCACHE en dinámico, la memoria caché de clúster puede expandirse según sea necesario. Sin embargo, primero debe asegurarse de que las salidas de carga de trabajo de clúster instaladas puedan funcionar con una memoria caché dinámica.

Si una salida de carga de trabajo de clúster instalada no puede funcionar con un mensaje de memoria caché dinámica, se emite CSQM061E .

Se proporciona MQXCLWLNpara que las salidas de carga de trabajo de clúster naveguen por la memoria caché de clúster de una forma que funcione independientemente de si se utilizan memorias caché dinámicas o estáticas.

Para los nuevos gestores de colas, establezca CLCACHE=DYNAMIC, a menos que vaya a utilizar una salida de carga de trabajo de clúster que no dé soporte a una memoria caché dinámica.

Para los gestores de colas existentes que ya utilizan una memoria caché estática y están en un clúster que no tiene muchas colas nuevas y gestores de colas añadidos, es razonable seguir utilizando CLCACHE=STATIC.

Para los gestores de colas existentes que ya utilizan una memoria caché estática y que están en un clúster al que se le van a añadir muchas colas o gestores de colas nuevos, empiece a utilizar CLCACHE=DYNAMIC.

STATIC
Cuando la memoria caché del clúster es estática, su tamaño se establece durante el inicio del gestor de colas, con un valor suficiente para el volumen actual de información del clúster más algo de espacio para la expansión. El tamaño no puede aumentar mientras el gestor de colas está activo. Este es el valor predeterminado.

DYNAMIC
Cuando la memoria caché del clúster es dinámica, el tamaño inicial asignado durante el inicio del gestor de colas se puede aumentar automáticamente si es necesario mientras el gestor de colas está activo.

CMDUSER
Especifica el ID de usuario predeterminado utilizado para las comprobaciones de seguridad de mandato. Este ID de usuario debe estar definido en el ESM (por ejemplo, RACF). Especifique un nombre de 1 a 8 caracteres alfanuméricos. El primer carácter debe ser alfabético.

El valor predeterminado es CSQOPR.

EXCLMSG
Especifica la exclusión de una lista de mensajes de error.

Esta lista es dinámica y se actualiza con el mandato SET SYSTEM.

El valor predeterminado es una lista vacía ( ).

Los mensajes se proporcionan sin el prefijo CSQ y sin el sufijo del código de acción (I-D-E-A). Por ejemplo, para excluir el mensaje CSQX500I, añada X500 a esta lista. Esta lista puede contener un máximo de 16 identificadores de mensaje.

Para cumplir los requisitos de inclusión en la lista, el mensaje tiene que emitirse tras un inicio normal de los espacios de direcciones MSTR o CHIN y empezar con uno de los siguientes caracteres: E, H, I, J, L, M, N, P, R, T, V, W, X, Y, 2 ,3, 5, 9.

Los identificadores de mensaje que se emiten como resultado del procesamiento de mandatos, se pueden añadir a la lista, sin embargo, no se excluirán. Por ejemplo, se emite un identificador de mensaje como resultado del mandato DISPLAY USAGE PSID(*), sin embargo, este mensaje no se puede suprimir.

EXITLIM

Especifica el tiempo, en segundos, permitido para cada invocación de los programa de salida del gestor de colas. (Este parámetro no tiene ningún efecto en los programas de salida de canal.)

Especifique un valor comprendido entre 5 y 9999.

El valor predeterminado es 30. El gestor de colas sondea cada 30 segundos los programas de salida en ejecución. En cada sondeo, los programas de salida que se han ejecutado durante más tiempo que el especificado por EXITLIM se concluyen de forma forzosa.

EXITTCB

Especifica el número de tareas de servidor iniciadas que se deben utilizar para ejecutar programas de salidas en el gestor de colas. (Este parámetro no tiene ningún efecto en los programas de salida de canal.) Debe especificar como mínimo un número igual al número máximo de programas de salida (que no sean programas de salida de canal) que el gestor de colas pueda necesitar ejecutar. De lo contrario, se producirá una terminación anómala 6c6.

Especifique un valor comprendido entre cero y 99. Si se especifica cero, no se puede ejecutar ningún programa de salida.

El nivel predeterminado es 8.

LOGLOAD

Especifica el número de registros de anotaciones que IBM MQ graba entre el inicio de un punto de comprobación y el siguiente. IBM MQ inicia un nuevo punto de comprobación después de que se hayan escrito un número de registros especificados.

Especifique un valor comprendido entre 200 y 16.000.000.

El valor predeterminado es 500 000.

Cuanto mayor sea el valor, mejor será el rendimiento de IBM MQ, pero el reinicio necesitará más tiempo si el parámetro se establece en un valor elevado.

Valores sugeridos:

Sistema de prueba	10 000
Sistema de producción	500 000 En un sistema de producción, el valor predeterminado proporcionado puede dar como resultado una frecuencia de comprobación demasiado alta.

El valor de LOGLOAD determina la frecuencia de los puntos de comprobación del gestor de colas. Un valor demasiado alto hace que se escriba un gran volumen de datos en el archivo de registro entre puntos de comprobación, lo que da como resultado un mayor tiempo de reinicio de recuperación del gestor de colas después de un error. Un valor demasiado pequeño hace que los puntos de comprobación se produzcan con demasiada frecuencia durante los periodos de mayor carga de trabajo, lo cual afecta negativamente a los tiempos de respuesta y a la utilización del procesador.

Es recomendable un valor inicial de 500 000 para LOGLOAD. Para obtener una tasa de mensajes permanentes de 1 KB de 100 mensajes por segundo (es decir, 100 operaciones MQPUT con confirmación y 100 operaciones MQGET con confirmación), el intervalo entre puntos es aproximadamente 5 minutos.

Nota: Esto es sólo una directriz; el valor óptimo de este parámetro depende de las características de cada sistema.

MULCCAPT
Especifica el algoritmo que se debe utilizar para recopilar los datos utilizados por Measured Usage License Charging (MULC).

ESTÁNDAR
MULC se basa en el intervalo entre la llamada MQCONN de la API de IBM MQ y la llamada MQDISC de la API de IBM MQ.

REFINED
MULC se basa en el intervalo entre el inicio de una llamada de API de IBM MQ y el final de la llamada de API de IBM MQ.

El valor predeterminado es STANDARD

OTMACON
Parámetros OTMA. Esta palabra clave tiene cinco parámetros posicionales:

OTMACON = (`Group,Member,Druexit,Age,Tpipepfx`)

Grupo
Este es el nombre del grupo XCF al que pertenece esta instancia concreta de IBM MQ.

Puede tener de 1 a 8 caracteres de longitud y se debe escribir en mayúsculas.

El valor predeterminado son espacios en blanco, lo que indica que IBM MQ no debe intentar unirse a un grupo XCF.

Miembro
Este es el nombre de miembro de esta instancia determinada de IBM MQ dentro del grupo XCF.

Puede tener de 1 a 16 caracteres de longitud y se debe escribir en mayúsculas.

El valor predeterminado es el nombre de gestor de colas de 4 caracteres.

Druexit
Especifica el nombre de la salida de usuario de resolución de destino OTMA que debe ser ejecutada por IMS.

Puede tener de 1 a 8 caracteres de longitud.

El valor predeterminado es DFSYDRU0.

Este parámetro es opcional; es necesario si IBM MQ debe recibir mensajes de una aplicación IMS que no ha sido iniciada por IBM MQ. El nombre debe corresponder a la salida de usuario de resolución de destino codificada en el sistema IMS. Para más información, consulte "Using OTMA exits in IMS" en la página 1015.

Age
Representa el periodo de tiempo, en segundos, durante el que un ID de usuario de IBM MQ se considera que ha sido verificado previamente por IMS.

Su valor está comprendido entre cero y 2 147 483 647.

El valor predeterminado es 2 147 483 647.

Se recomienda que establezca este parámetro junto con el parámetro `interval` del mandato ALTER SECURITY para mantener la coherencia de los valores de memoria caché de seguridad en el sistema principal.

Tpipepfx
Especifica el prefijo que se debe utilizar para los nombres de Tpipe.

Consta de tres caracteres; el primer carácter está en el rango de la A a la Z, los caracteres siguientes van de la A a la Z o del 0 al 9. El valor predeterminado es CSQ.

Se utiliza cada ve que IBM MQ crea un Tpipe; el resto del nombre lo asigna IBM MQ. El usuario no puede establecer el nombre de Tpipe completo para ningún Tpipe creado por IBM MQ.

QINDXBLD
Determina si el reinicio del gestor de colas espera a que estén reconstruidos todos los índices de cola, o el reinicio se completa antes de que estén reconstruidos todos los índices.

WAIT
El reinicio del gestor de colas espera a que finalice la creación de todos los índices de cola. Esto significa que no se retarda ninguna aplicación durante el proceso normal de la API de IBM MQ mientras se crea el índice, pues todos los índices se crean antes de que las aplicaciones se puedan conectar al gestor de colas.

Este es el valor predeterminado.

NOWAIT
El gestor de colas se puede reiniciar antes de que finalice la creación de todos los índice de cola.

QMCCSID
Especifica el identificador de juego de caracteres codificados predeterminado que debe utilizar el gestor de colas (y por tanto, la gestión de colas distribuidas).

Especifique un valor comprendido entre cero y 65535. El valor debe representar una página de códigos EBCDIC listada como una página de códigos nativa de z/OS para el idioma elegido en Idiomas nacionales.

Cero, que es el valor predeterminado, significa utilizar el CCSID establecido actualmente o, si no hay ninguno establecido, utilizar el CCSID 500. Esto significa que si ha establecido explícitamente el CCSID en un valor cualquiera distinto de cero, no puede redefinir el valor estableciendo QMCCSID en cero; ahora debe utilizar el CCSID correcto distinto de cero. Si QMCCSID es cero, puede comprobar qué CCSID se utiliza realmente emitiendo el mandato DISPLAR QMGR CCSID.

Nota: Todos los gestores de colas de un grupo de compartición de colas deben utilizar el mismo QMCCSID.

QSGDATA
Datos del grupo de compartición de colas. Esta palabra clave tiene cinco parámetros posicionales:

QSGDATA=(Qsgname,Dsgname,Db2name,Db2serv ,Db2blob)

Qsgname
Especifica el nombre del grupo de compartición de colas al que pertenece el gestor de colas.

Consulte Reglas de denominación de objetos de IBM MQ para obtener caracteres válidos. El nombre:

- Puede tener de 1 a 4 caracteres de longitud
- No debe comenzar con un valor numérico
- No debe acabar en @.

  Esto se debe a que, por razones de implementación, los nombres con menos de cuatro caracteres se rellenan internamente con símbolos @.

El valor predeterminado es espacios en blanco, lo que indica que el gestor de colas no es miembro de ningún grupo de compartición de colas.

Dsgname
Es el nombre del grupo de compartición de datos de Db2 al que se va a conectar el gestor de colas.

Puede tener de 1 a 8 caracteres de longitud y se debe escribir en mayúsculas.

El valor predeterminado es espacios en blanco, lo que indica que no está utilizando grupos de compartición de colas.

Db2name
Es el nombre del subsistema o la conexión de grupo de Db2 al que se va a conectar el gestor de colas.

Puede tener de 1 a 4 caracteres de longitud y se debe escribir en mayúsculas.

El valor predeterminado es espacios en blanco, lo que indica que no está utilizando grupos de compartición de colas.

Nota: El subsistema (o conexión de grupo) de Db2 debe estar en el grupo de compartición de datos de Db2 especificado en `Dsgname`, y todos los gestores de colas deben especificar el mismo grupo de compartición de datos de Db2.

Db2serv
Es el número de tareas de servidor que se utilizan para acceder a Db2.

Este valor está comprendido entre 4 y 10.

El valor predeterminado es 4.

Db2blob
Es el número de tareas de Db2 que se utilizan para acceder a Objetos binarios grandes (BLOB).

Este valor está comprendido entre 4 y 10.

El valor predeterminado es 4.

Si especifica uno de los parámetros de nombre (es decir, **Qsgname**, **Dsgname** o **Db2name**), debe especificar valores para los otros nombres, de lo contrario IBM MQ falla.

RESAUDIT
Especifica si se graban registros de auditoría RACF para las comprobaciones de seguridad RESLEVEL que se realizan durante el proceso de conexión.

Especifique uno de los valores siguientes:

NO
No se realiza auditoría RESLEVEL.

SÍ
Se realiza auditoría RESLEVEL.

El valor predeterminado es YES.

ROUTCDE
Especifica el código de direccionamiento de mensajes predeterminado de z/OS que se asigna a los mensajes que no se envían en respuesta directa a un mandato MQSC.

Especifique uno de los valores siguientes:

1. Un valor comprendido entre 1 y 16 inclusive.
2. Una lista de valores, separados por una coma y encerrados entre paréntesis. Cada valor debe estar comprendido entre 1 y 16 inclusive.

El valor predeterminado es 1.

Para obtener más información sobre los códigos de direccionamiento de z/OS , consulte *Códigos de direccionamiento* en *Descripción de mensaje* en uno de los volúmenes de los manuales de *z/OS MVS System Messages* .

SERVICIO
Este campo está reservado para uso de IBM.

SMFACCT
Especifica si IBM MQ envía los datos contables a SMF automáticamente cuando se inicia el gestor de colas.

Especifique uno de los valores siguientes:

NO
No iniciar automáticamente la recopilación de datos contables.

SÍ
Iniciar automáticamente la recopilación de datos contables para la clase predeterminada 1.

integers
Una lista de clases para las que los datos de contabilidad se recopilan automáticamente en el rango de 1 a 4.

* Inicie automáticamente la contabilidad SMF para las clases 1, 2 y 3.

El valor predeterminado es NO.

SMFSTAT
Especifica si se deben recoger estadísticas SMF automáticamente cuando se inicia el gestor de colas.

Especifique uno de los valores siguientes:

NO
No iniciar automáticamente la recopilación de estadísticas.

SÍ
Iniciar automáticamente la recopilación de estadísticas para la clase predeterminada 1.

integers
Una lista de clases para las que se recopilan estadísticas automáticamente en el rango de 1 a 5.

Para recopilar estadísticas de clase 2 o 3, la clase 1 también se debe especificar.

* Inicie automáticamente las estadísticas SMF para las clases 1, 2 y 3.

El valor predeterminado es NO.

SPLCAP
La función de política de seguridad habilita un nivel superior de seguridad de mensajes mediante las políticas que controlan si los mensajes se firman o cifran a medida que se graban y leen desde las colas.

El proceso de política de seguridad se configura para este gestor de colas, estableciendo SPLCAP en uno de los valores siguientes:

NO
La función de implementar políticas de seguridad de mensajes para colas no está habilitada durante la inicialización del gestor de colas.

SÍ
Las funciones de seguridad de mensajes se habilitan durante la inicialización del gestor de colas.

El gestor de colas comprueba que el atributo AMSPROD esté establecido en uno de AMS, ADVANCED o ADVANCEDVUE, en cuyo caso se ha licenciado para AMS. De lo contrario, no se iniciará.

El gestor de colas también comprueba si la configuración de AMS necesaria está en vigor. Si no es así, el gestor de colas no se iniciará.

Si el gestor de colas tiene licencia para AMSy la configuración necesaria está en su lugar, el gestor de colas empezará con las prestaciones de seguridad de mensajes habilitadas durante la inicialización del gestor de colas y se iniciará el espacio de direcciones AMSM.

El valor predeterminado es NO.

STATIME
A partir de IBM MQ for z/OS 9.3.0, especifica el tiempo, en minutos y segundos, entre las recopilaciones consecutivas de datos estadísticos. Si ACCTIME no se ha establecido, o es `-1`, también especifica el tiempo entre recopilaciones consecutivas de datos contables.

Especifique un número en el rango de `0` a 1440 minutos en el formato `'mmmm'`, o en el rango de `0` a 1440 minutos, y de `0` a 59 segundos, en el formato `'mmmm.ss'`. El valor predeterminado es 30 minutos.

Notas:

- Cuando solo se especifica un intervalo de segundos, se debe añadir al intervalo un valor de `0` como prefijo. El intervalo más pequeño posible es un segundo: `'0.01'`.
- A partir de IBM MQ for z/OS 9.3.0, si especifica un valor de `0`, los datos estadísticos se recopilan en la difusión de recopilación de datos de SMF. Si no se especifica ACCTIME, o si es `-1`, también se recopilan datos contables en la difusión de recopilación de datos de SMF. Consulte Utilización de System Management Facility si desea más información.
- Si especifica un valor de `-1`, que es el valor predeterminado, los datos contables se recopilan en el intervalo especificado por el valor STATIME.

TRACSTR
Especifica si se debe iniciar automáticamente el rastreo global.

Especifique uno de los valores siguientes:

NO
No iniciar automáticamente el rastreo global.

SÍ
Iniciar automáticamente el rastreo global para la clase predeterminada 1.

integers
Lista de clases para las que se debe iniciar automáticamente el rastreo global, dentro del rango del 1 al 4.

*****
Iniciar automáticamente el rastreo global para todas las clases.

El valor predeterminado es NO si no especifica la palabra clave en la macro.

Nota: El módulo de carga de parámetros del sistema predeterminado proporcionado (CSQZPARM) tiene TRACSTR=YES (establecido en el módulo de ensamblador CSQFSYSP). Si no desea iniciar el rastreo automáticamente, puede crear su propio módulo del parámetro del sistema, o emitir el mandato STOP TRACE cuando el gestor de colas se haya iniciado.

Para obtener información detallada sobre el mandato STOP TRACE, consulte STOP TRACE.

TRACTBL
Especifica el tamaño predeterminado, en bloques de 4 KB, de la tabla de rastreo donde el recurso de rastreo global almacena los registros de rastreo de IBM MQ.

Especifique un valor comprendido entre 1 y 999.

El valor predeterminado es 99. Esto equivale a 396 KB.

Nota: El almacenamiento para la tabla de rastreo se asigna en el ECSA. Por consiguiente, debe seleccionar este valor con cuidado.

WLMTIME
Especifica el tiempo (en minutos o segundos, dependiendo del valor de WLMTIMU) que transcurre entre las exploraciones de índices para colas gestionadas por WLM.

Especifique un valor comprendido entre 1 y 9999.

El valor predeterminado es 30.

WLMTIMU
Unidades de tiempo utilizadas con el parámetro WLMTIME.

Especifique uno de los valores siguientes:

MINS
WLMTIME representa un número de minutos.

SECS
WLMTIME representa un número de segundos.

El valor predeterminado es MINS.

Referencia relacionada

"Using CSQ6LOGP" en la página 935
Use this topic as a reference for how to specify logging options using CSQ6LOGP.

"Using CSQ6ARVP" en la página 939
Use this topic as a reference for how to specify your archiving environment using CSQ6ARVP

z/OS *Using CSQ6LOGP*

Use this topic as a reference for how to specify logging options using CSQ6LOGP.

Use CSQ6LOGP to establish your logging options.

The default parameters for CSQ6LOGP, and whether you can alter each parameter using the SET LOG command, are shown in Default values of CSQ6LOGP parameters. If you need to change any of these values, refer to the detailed descriptions of the parameters.

Table 61. Default values of CSQ6LOGP parameters

Parameter	Description	Default value	SET command
COMPLOG	Controls whether log compression is enabled.	NONE	X
DEALLCT	Length of time an archive tape unit remains unused before it is deallocated.	zero	X
INBUFF	Size of input buffer storage for active and archive log data sets.	60 KB	-
MAXARCH	Maximum number of archive log volumes that can be recorded.	500	X
MAXCNOFF	Maximum number of CSQJOFF7 offload tasks that can be run in parallel.	31	-
MAXRTU	Maximum number of dedicated tape units allocated to read archive log tape volumes concurrently.	2	X
OFFLOAD	Archiving on or off.	YES (ON)	-
OUTBUFF	Size of output buffer storage for active and archive log data sets.	4 000 KB	-
TWOACTV	Single or dual active logging.	YES (dual)	-
TWOARCH	Single or dual archive logging.	YES (dual)	-
TWOBSDS	Single or dual BSDS.	YES (dual BSDS)	-
WRTHRSH	Number of output buffers to be filled before they are written to the active log data sets.	20	X
ZHYWRITE	Specifies whether the zHyperWrite feature is enabled.	NO	X
V 9.4.0 ZHYLINK	Specifies whether the zHyperLink feature is enabled.	NO	X

COMPLOG

Specifies whether log compression is enabled.

Specify either:

NONE

Log compression is not enabled.

RLE
Log compression is enabled using run-length encoding.

ANY
The queue manager selects the compression algorithm that gives the greatest degree of log record compression. This option results in RLE compression.

The default is NONE.

For more details about log compression, see Log compression.

DEALLCT
Specifies the length of time, in minutes, that an archive read tape unit is allowed to remain unused before it is deallocated.

Specify one of the following:

- Time, in minutes, in the range zero through 1440
- NOLIMIT

Specifying 1440 or NOLIMIT means that the tape unit is never deallocated.

The default is zero.

When archive log data is being read from tape, it is recommended that you set this value high enough to allow IBM MQ to optimize tape handling for multiple read applications.

INBUFF
Specifies the size, in kilobytes, of the input buffer for reading the active and archive logs during recovery. Use a decimal number in the range 28 through 60. The value specified is rounded up to a multiple of 4.

The default is 60 KB.

Suggested settings:

Test system	28 KB
Production system	60 KB Set this to the maximum for best log read performance.

MAXARCH
Specifies the maximum number of archive log volumes that can be recorded in the BSDS. When this number is exceeded, recording begins again at the start of the BSDS.

Use a decimal number in the range 10 through 1000.

The default is 500.

Suggested settings:

Test system	500 (default)
Production system	1 000 Set this to the maximum so that the BSDS can record as many logs as possible.

For information about the logs and BSDS, see Managing IBM MQ resources.

MAXCNOFF
Specifies the number of CSQJOFF7 offload tasks that can be run in parallel.

This allows a queue manager, or queue managers, to be tuned such that they will not use all the available tape units.

Instead the queue manager waits until a CSQJOFF7 offload task has completed before trying to allocate any new archive data sets.

If the queue manager is archiving to tape, set this parameter so that the number of concurrent tape requests should not equal, or exceed, the number of tape units available, otherwise the system might hang.

Note that if dual archiving is in use, then each offload task performs both archives, so the parameter needs to be set accordingly. For example if the queue manager is dual archiving to tape, a value of MAXCNOFF=2 would allow up to two active logs to be archived concurrently to four tapes.

If several queue managers are sharing the tape units, you should set the MAXCNOFF for each queue manager accordingly.

The default value is 31.

Specify a value in the range 1 through 31.

MAXRTU
Specifies the maximum number of dedicated tape units that can be allocated to read archive log tape volumes concurrently.

This parameter and the DEALLCT parameter allow IBM MQ to optimize archive log reading from tape devices.

Specify a value in the range 1 through 99.

The default is 2.

It is recommended that you set the value to be at least one less than the number of tape units available to IBM MQ. If you do otherwise, the offload process could be delayed, which could affect the performance of your system. For maximum throughput during archive log processing, specify the largest value possible for this option, remembering that you need at least one tape unit for offload processing.

OFFLOAD
Specifies whether archiving is on or off.

Specify either:

YES
Archiving is on

NO
Archiving is off

The default is YES.

Attention: Do **not** switch archiving off unless you are working in a test environment. If you do switch it off, you cannot guarantee that data will be recovered in the event of a system or transaction failure.

OUTBUFF
Specifies the total size, in kilobytes, of the storage to be used by IBM MQ for output buffers for writing the active and archive log data sets. Each output buffer is 4 KB.

The parameter must be in the range 128 through 4000. The value specified is rounded up to a multiple of 4. Values between 40 and 128 will be accepted for compatibility reasons, and are treated as a value of 128.

The default is 4000 KB.

Suggested settings:

Test system	400 KB
Production system	4 000 KB Set this value to the maximum to avoid running out of log output buffers.

TWOACTV
Specifies single or dual active logging.

Specify either:

NO
Single active logs

YES
Dual active logs

The default is YES.

For more information about the use of single and dual logging, see Managing IBM MQ resources.

TWOARCH
Specifies the number of archive logs that IBM MQ produces when the active log is offloaded.

Specify either:

NO
Single archive logs

YES
Dual archive logs

The default is YES.

Suggested settings:

Test system	NO
Production system	YES (default)

For more information about the use of single and dual logging, see Managing IBM MQ resources.

TWOBSDS
Specifies the number of bootstrap data sets.

Specify either:

NO
Single BSDS

YES
Dual BSDS

The default is YES.

For more information about the use of single and dual logging, see Managing IBM MQ resources.

WRTHRSH
Specifies the number of 4 KB output buffers to be filled before they are written to the active log data sets.

The larger the number of buffers, the less often the write takes place, and this improves the performance of IBM MQ. The buffers might be written before this number is reached if significant events, such as a commit point, occur.

Specify the number of buffers in the range 1 through 256.

The default is 20.

ZHYWRITE
Especifica si están habilitando las grabaciones en los registros activos realizadas con zHyperWrite.

Para obtener más información, consulte Utilización de zHyperWrite con registros activos de IBM MQ.

El valor puede ser:

NO
zHyperWrite no se ha habilitado.

SÍ
zHyperWrite está habilitado.

V 9.4.0 **ZHYLINK**
Specifies whether writes to the active logs are made with zHyperLink being enabled.

For more information on enabling active logs with zHyperLink, see Using zHyperLink with IBM MQ.

The value can be:

NO
zHyperLink is not enabled.

YES
zHyperLink is enabled.

Note: Enabling ZHYLINK also enables ZHYWRITE

Related reference

"Utilización de CSQ6SYSP" on page 925
Utilice este tema como referencia para conocer cómo establecer parámetros del sistema utilizando CSQ6SYSP.

"Using CSQ6ARVP" on page 939
Use this topic as a reference for how to specify your archiving environment using CSQ6ARVP

z/OS *Using CSQ6ARVP*
Use this topic as a reference for how to specify your archiving environment using CSQ6ARVP

Use CSQ6ARVP to establish your archiving environment.

The default parameters for CSQ6ARVP, and whether you can alter each parameter using the SET ARCHIVE command, are shown in Table 62 on page 939. If you need to change any of these values, refer to the detailed descriptions of the parameters. For more information about planning your storage, see Planning your storage and performance requirements on z/OS .

Table 62. Default values of CSQ6ARVP parameters

Parameter	Description	Default value	SET command
ALCUNIT	Units in which primary and secondary space allocations are made.	BLK (blocks)	X
ARCPFX1	Prefix for first archive log data set name.	CSQARC1	X
ARCPFX2	Prefix for second archive log data set name.	CSQARC2	X
ARCRETN	The retention period of the archive log data set in days.	9999	X
ARCWRTC	List of route codes for messages to the operator about archive log data sets.	1,3,4	X
ARCWTOR	Whether to send message to operator and wait for reply before trying to mount an archive log data set.	YES	X
BLKSIZE	Block size of archive log data set.	28 672	X
CATALOG	Whether archive log data sets are cataloged in the ICF.	NO	X
COMPACT	Whether archive log data sets should be compacted.	NO	X
PRIQTY	Primary space allocation for DASD data sets.	25 715	X

Table 62. Default values of CSQ6ARVP parameters (continued)			
Parameter	**Description**	**Default value**	**SET command**
PROTECT	Whether archive log data sets are protected by ESM profiles when the data sets are created.	NO	X
QUIESCE	Maximum time, in seconds, allowed for quiesce when ARCHIVE LOG with MODE(QUIESCE) specified.	5	X
SECQTY	Secondary space allocation for DASD data sets. See the ALCUNIT parameter for the units to be used.	540	X
TSTAMP	Whether the archive data set name should include a time stamp.	NO	X
UNIT	Device type or unit name on which the first copy of archive log data sets is stored.	TAPE	X
UNIT2	Device type or unit name on which the second copy of archive log data sets is stored.	Blank	X

ALCUNIT
Specifies the unit in which primary and secondary space allocations are made.

Specify one of:

CYL
Cylinders

TRK
Tracks

BLK
Blocks

You are recommended to use BLK because it is independent of the device type.

The default is BLK.

If free space on the archive DASD volumes is likely to be fragmented, you are recommended to specify a smaller primary extent and allow expansion into secondary extents. For more information about space allocation for active logs, refer to Planning your log archive storage.

ARCPFX1
Specifies the prefix for the first archive log data set name.

See the TSTAMP parameter for a description of how the data sets are named and for restrictions on the length of ARCPFX1.

This parameter cannot be left blank.

The default is CSQARC1.

You might need to authorize the userid associated with the IBM MQ queue manager address space to create archive logs with this prefix.

ARCPFX2
Specifies the prefix for the second archive log data set name.

See the TSTAMP parameter for a description of how the data sets are named and for restrictions on the length of ARCPFX2.

This parameter cannot be blank even if the TWOARCH parameter is specified as NO.

The default is CSQARC2.

You might need to authorize the userid associated with the IBM MQ queue manager address space to create archive logs with this prefix.

ARCRETN
Specifies the retention period, in days, to be used when the archive log data set is created.

The parameter must be in the range zero through 9999.

The default is 9999.

Suggested settings:

Test system	3 In a test system, archive logs are probably not required over long periods.
Production system	9 999 (default) Set this value high to effectively switch automatic archive log deletion off.

For more information about discarding archive log data sets, see Discarding archive log data sets.

ARCWRTC
Specifies the list of z/OS routing codes for messages about the archive log data sets to the operator. This field is ignored if ARCWTOR is set to NO.

Specify up to 14 routing codes, each with a value in the range 1 through 16. You must specify at least one code. Separate codes in the list by commas, not by blanks.

The default is the list of values: 1,3,4.

For more information about z/OS routing codes, see *Routing codes* in *Message description* in one of the volumes of the *z/OS MVS System Messages* manuals.

ARCWTOR
Specifies whether a message is to be sent to the operator and a reply is received before attempting to mount an archive log data set.

Other IBM MQ users might be forced to wait until the data set is mounted, but they are not affected while IBM MQ is waiting for the reply to the message.

Specify either:

YES
The device needs a long time to mount archive log data sets. For example, a tape drive.

NO
The device does not have long delays. For example, DASD.

The default is YES.

Suggested settings:

Test system	NO
Production system	YES (default) This is dependent on operational procedures. If tape robots are used, NO might be more appropriate.

BLKSIZE
Specifies the block size of the archive log data set. The block size you specify must be compatible with the device type you specify in the UNIT parameter.

The parameter must be in the range 4 097 through 28 672. The value you specify is rounded up to a multiple of 4 096.

The default is 28 672.

This parameter is overridden by the storage management subsystem (SMS) data class blocksize, if it is provided.

If the archive log data set is written to DASD, you are recommended to choose the maximum block size that allows two blocks for each track. For example, for a 3390 device, you should use a block size of 24 576.

If the archive log data set is written to tape, specifying the largest possible block size improves the speed of reading the archive log. You should use a block size of 28 672.

Suggested settings:

Test system	Use the block size recommendation depending on the media used for archive logs. That is, for disk 24 576, and tape 28 672.
Production system	Use the block size recommendation depending on the media used for archive logs. That is, for disk 24 576, and tape 28 672.

CATALOG
Specifies whether archive log data sets are cataloged in the primary integrated catalog facility (ICF) catalog.

Specify either:

NO
Archive log data sets are not cataloged

YES
Archive log data sets are cataloged

The default is NO.

All archive log data sets allocated on DASD must be cataloged. If you archive to DASD with the CATALOG parameter set to NO, message CSQJ072E is displayed each time an archive log data set is allocated, and IBM MQ catalogs the data set.

Suggested settings:

Test system	YES
Production system	YES, when archives are allocated on DASD

COMPACT
Specifies whether data written to archive logs is to be compacted. This option applies only to a 3480 or 3490 device that has the improved data recording capability (IDRC) feature. When this feature is turned on, hardware in the tape control unit writes data at a much higher density than normal, allowing for more data on each volume. Specify NO if you do not use a 3480 device with the IDRC feature or a 3490 base model, except for the 3490E. Specify YES if you want the data to be compacted.

Specify either:

NO
Do not compact the data sets

YES
Compact the data sets

The default is NO.

Specifying YES adversely affects performance. Also be aware that data compressed to tape can be read only using a device that supports the IDRC feature. This can be a concern if you have to send archive tapes to another site for remote recovery.

Suggested settings:

Test system	Not applicable
Production system	NO (default) This applies to 3480 and 3490 IDR compression only. Setting this to YES might degrade archive log read performance during recovery and restart; however, it does not affect writing to tape.

PRIQTY

Specifies the primary space allocation for DASD data sets in ALCUNITs.

The value must be greater than zero.

The default is 25 715.

This value must be sufficient for a copy of either the log data set or its corresponding BSDS, whichever is the larger. To determine the necessary value, follow this procedure:

1. Determine the number of active log records allocated (c) as explained in "Create the bootstrap and log data sets" on page 921.
2. Determine the number of 4096 byte blocks in each archive log block:

   ```
   d = BLKSIZE / 4096
   ```

   where BLKSIZE is the rounded up value.
3. If ALCUNIT=BLK:

   ```
   PRIQTY = INT(c / d) + 1
   ```

   where INT means round down to an integer.

   If ALCUNIT=TRK:

   ```
   PRIQTY = INT(c / (d * INT(e/BLKSIZE))) + 1
   ```

   where e is the number of bytes for each track (56664 for a 3390 device) and INT means round down to an integer.

   If ALCUNIT=CYL:

   ```
   PRIQTY = INT(c / (d * INT(e/BLKSIZE) * f)) + 1
   ```

   where f is the number of tracks for each cylinder (15 for a 3390 device) and INT means round down to an integer.

For information about how large to make your log and archive data sets, see "Create the bootstrap and log data sets" on page 921 and "Define your page sets" on page 922.

Suggested settings:

Test system	1 680 Sufficient to hold the entire active log, that is: `10  080 / 6 = 1  680 blocks`
Production system	Not applicable when archiving to tape.

If free space on the archive DASD volumes is likely to be fragmented, you are recommended to specify a smaller primary extent and allow expansion into secondary extents. For more information about space allocation for active logs, see Planning your log archive storage.

PROTECT
Specifies whether archive log data sets are to be protected by discrete ESM (external security manager) profiles when the data sets are created.

Specify either:

NO
Profiles are not created.

YES
Discrete data set profiles are created when logs are offloaded. If you specify YES:

- ESM protection must be active for IBM MQ.
- The user ID associated with the IBM MQ queue manager address space must have authority to create these profiles.
- The TAPEVOL class must be active if you are archiving to tape.

Otherwise, offloading fails.

The default is NO.

QUIESCE
Specifies the maximum time in seconds allowed for the quiesce when an ARCHIVE LOG command is issued with MODE(QUIESCE) specified.

The parameter must be in the range 1 through 999.

The default is 5.

SECQTY
Specifies the secondary space allocation for DASD data sets in ALCUNITs. The secondary extent can be allocated up to 15 times; see the IBM z/OS Management Facility Programming Guide for more information on ALCUNIT.

The parameter must be greater than zero.

The default is 540.

TSTAMP
Specifies whether the archive log data set name has a time stamp in it.

Specify either:

NO
Names do not include a time stamp. The archive log data sets are named:

```
arcpfxi.A nnnnnnn
```

Where *arcpfxi* is the data set name prefix specified by ARCPFX1 or ARCPFX2. *arcpfxi* can have up to 35 characters.

YES
Names include a time stamp. The archive log data sets are named:

```
arcpfxi.cyyddd.T hhmmsst.A nnnnnnn
```

where *c* is 'D' for the years up to and including 1999 or 'E' for the year 2000 and later, and *arcpfxi* is the data set name prefix specified by ARCPFX1 or ARCPFX2. *arcpfxi* can have up to 19 characters.

EXT
Names include a time stamp. The archive log data sets are named:

```
arcpfxi.D yyyyddd.T hhmmsst.A nnnnnnn
```

Where *arcpfxi* is the data set name prefix specified by ARCPFX1 or ARCPFX2. *arcpfxi* can have up to 17 characters.

The default is NO.

UNIT
Specifies the device type or unit name of the device that is used to store the first copy of the archive log data set.

Specify a device type or unit name of 1 through 8 alphanumeric characters. The first character must be alphabetic.

This parameter cannot be blank.

The default is TAPE.

If you archive to DASD, you can specify a generic device type with a limited volume range, for example, UNIT=3390.

If you archive to DASD, make sure that:

- The primary space allocation is large enough to contain all the data from the active log data sets.
- The archive log data set catalog option (CATALOG) is set to YES.
- You have used a proper value for BLKSIZE.

If you archive to TAPE, IBM MQ can extend to a maximum of 20 volumes.

Suggested settings:

Test system	DASD
Production system	TAPE

For more information about choosing a location for archive logs, see Planning your log archive storage.

UNIT2
Specifies the device type or unit name of the device that is used to store the second copy of the archive log data sets.

Specify a device type or unit name of 1 through 8 alphanumeric characters. The first character must be alphabetic. If this parameter is blank, the value set for the UNIT parameter is used.

The default is blank.

Related reference
"Utilización de CSQ6SYSP" on page 925
Utilice este tema como referencia para conocer cómo establecer parámetros del sistema utilizando CSQ6SYSP.

"Using CSQ6LOGP" on page 935
Use this topic as a reference for how to specify logging options using CSQ6LOGP.

z/OS *Using CSQ6USGP*

Use this topic as a reference for how to set your system parameters using CSQ6USGP

Use CSQ6USGP to control product usage recording.

The default parameters for CSQ6USGP are shown in Table 63 on page 946. If you need to change any of these values, refer to the detailed descriptions of the parameters.

Attention: You cannot alter any of these parameters using the SET SYSTEM command.

Table 63. Default values of CSQ6USGP parameters

Parameter	Description	Default value
QMGRPROD	Product against which queue manager usage is to be recorded	Blank
AMSPROD	Product against which Advanced Message Security (AMS) usage is to be recorded	Blank

QMGRPROD

Specifies the product against which queue manager usage is to be recorded.

Specify one of:

MQ

Queue manager usage is recorded as a stand-alone IBM MQ for z/OS product, with product ID 5655-MQ9.

VUE

Queue manager usage is recorded as a stand-alone IBM MQ for z/OS Value Unit Edition (VUE) product, with product ID 5655-VU9.

ADVANCEDVUE

Queue manager usage is recorded as part of an IBM MQ Advanced for z/OS Value Unit Edition product, with product ID 5655-AV1.

AMSPROD

If this parameter is not set the AMS address space will not start up and message CSQY024I will be output.

Specifies the product against which Advanced Message Security usage is to be recorded, if used.

Specify one of:

AMS

AMS usage is recorded as a stand-alone Advanced Message Security for z/OS product, with product ID 5655-AM9.

ADVANCED

AMS usage is recorded as part of an IBM MQ Advanced for z/OS product, with product ID 5655-AV9.

ADVANCEDVUE

AMS usage is recorded as part of an IBM MQ Advanced for z/OS Value Unit Edition product, with product ID 5655-AV1.

See Reporting product information for more information on product usage recording.

Related reference

“Utilización de CSQ6SYSP” on page 925

Utilice este tema como referencia para conocer cómo establecer parámetros del sistema utilizando CSQ6SYSP.

“Using CSQ6LOGP” on page 935

Use this topic as a reference for how to specify logging options using CSQ6LOGP.

z/OS *Tailor the channel initiator parameters*

Use ALTER QMGR to customize the channel initiator to suit your requirements.

- *Repeat this task for each IBM MQ queue manager, as required.*
- *You must perform this task when migrating from a previous version.*

A number of queue manager attributes control how distributed queuing operates. Set these attributes using the MQSC command ALTER QMGR. The initialization data set sample thlqual.SCSQPROC(CSQ4INYG) contains some settings that you can customize. For more information, see ALTER QMGR.

The values of these parameters are displayed as a series of messages each time you start the channel initiator.

The relationship between adapters, dispatchers, and maximum number of channels

The ALTER QMGR parameters CHIADAPS and CHIDISPS define the number of task control blocks (TCBs) used by the channel initiator. CHIADAPS (adapter) TCBs are used to make IBM MQ API calls to the queue manager. CHIDISPS (dispatcher) TCBs are used to make calls to the communications network.

The ALTER QMGR parameter MAXCHL influences the distribution of channels over the dispatcher TCBs.

CHIDISPS

If you have a small number of channels use the default value.

One task for each processor optimizes system performance. As dispatcher tasks are CPU intensive, the principle is to keep as few tasks as busy as possible, so that the time taken to find and start threads is minimized.

CHIDISPS(20) is suitable for systems with more than 100 channels. There is unlikely to be any significant disadvantage in having CHIDISPS(20) where this is more dispatcher TCBs than necessary.

As a guideline, if you have more than 1000 channels, allow one dispatcher for every 50 current channels. For example, specify CHIDISPS(40) to handle up to 2000 active channels.

If you are using TCP/IP, the maximum number of dispatchers used for TCP/IP channels is 100, even if you specify a larger value in CHIDISPS.

CHIADAPS

Each IBM MQ API call to the queue manager is independent of any other and can be made on any adapter TCB. Calls using persistent messages can take much longer than those for nonpersistent messages because of log I/O. Thus a channel initiator processing a large number of persistent messages across many channels may need more than the default 8 adapter TCBs for optimum performance. This is particularly so where achieved batchsize is small, because end of batch processing also requires log I/O, and where thin client channels are used.

The suggested value for a production environment is CHIADAPS(30). Using more than this is unlikely to give any significant extra benefit, and there is unlikely to be any significant disadvantage in having CHIADAPS(30) if this is more adapter TCBs than necessary.

MAXCHL

Each channel is associated with a particular dispatcher TCB at channel start and remains associated with that TCB until the channel stops. Many channels can share each TCB. MAXCHL is used to spread channels across the available dispatcher TCBs. The first (MIN((MAXCHL / CHIDISPS) , 10)) channels to start are associated with the first dispatcher TCB, and so on, until all dispatcher TCBs are in use.

The effect of this for small numbers of channels and a large MAXCHL is that channels are NOT evenly distributed across dispatchers. For example, if you set CHIDISPS(10) and left MAXCHL at its default value of 200 but had only 50 channels, five dispatchers would be associated with 10 channels each and five would be unused. We suggest setting MAXCHL to the number of channels actually to be used where this is a small fixed number.

If you change this queue manager property, you must also review the ACTCHL, LU62CHL, and TCPCHL queue manager properties to ensure that the values are compatible. See Queue manager parameters for a full description of these properties, and their relationship.

Setting up your z/OS UNIX System Services environment for channel initiators

The channel initiator (CHINIT) uses OMVS threads. Review the OMVS configuration parameters before creating a new CHINIT, or modifying the number of dispatchers or SSLTASKS.

Each CHINIT uses 3 + CHIDISP + SSLTASKS OMVS threads. These contribute to the total number of OMVS threads used in the LPAR, and towards the number of threads used by CHINIT started task user ID.

You can use the **D OMVS,L** and review the current usage, highwater usage, and system limit of MAXPROCSYS (the maximum number of processes that the system allows).

If you are adding a new CHINIT or increasing the values of CHIDISPS or SSLTASKS then you must calculate the increase in threads and review the impact on the MAXPROCSYS values. You can use the **SETOMVS** command to dynamically change the MAXPROCSYS, or update the BPXPRCxx parmlib value or both.

The OMVS parameter MAXPROCUSER is the number of OMVS threads a single OMVS user, that is with the same UID, can have. The threads count towards this value. So if you have 2 CHINITS with the same started task user ID, with 10 dispatchers and 3 SSLTASKS each then there are 2 *(3 +10 + 3) = 32 threads for the OMVS uid.

You can display the default MAXPROCUSER by issuing the **D OMVS,O** command and you can use the **SETOMVS** command to dynamically change the MAXPROCUSER, or update the BPXPRCxx parmlib value or both.

You can override this value on a per user basis with the RACF command **ALTUSER userid OMVS(PROCUSERMAX(nnnn))** or equivalent.

To start the channel initiator, issue the following command:

```
START CHINIT
```

To ensure that the channel initiator has started successfully, check that there is no ICH408I error in the xxxxCHIN(ssidCHIN) job log.

Related concepts

“Set up Batch, TSO, and RRS adapters” on page 948
Make the adapters available to applications by adding libraries to appropriate STEPLIB concatenations. To cater for SNAP dumps issued by an adapter, allocate a CSQSNAP DDname. Consider using CSQBDEFV to improve the portability of your application programs

Related reference

Channel initiator statistics data records

z/OS *Set up Batch, TSO, and RRS adapters*

Make the adapters available to applications by adding libraries to appropriate STEPLIB concatenations. To cater for SNAP dumps issued by an adapter, allocate a CSQSNAP DDname. Consider using CSQBDEFV to improve the portability of your application programs

- *Repeat this task for each IBM MQ queue manager as required.*
- *You might need to perform this task when migrating from a previous version.*

To make the adapters available to batch and other applications using batch connections, add the following IBM MQ libraries to the STEPLIB concatenation for your batch application :

- thlqual.SCSQANL *x*
- thlqual.SCSQAUTH

where *x* is the language letter for your national language. (You do not need to do this if the libraries are in the LPA or the link list.)

For TSO applications add the libraries to the STEPLIB concatenation in the TSO logon procedure or activate them using the TSO command TSOLIB.

If the adapter detects an unexpected IBM MQ error, it issues an z/OS SNAP dump to DDname CSQSNAP, and issues reason code MQRC_UNEXPECTED_ERROR to the application. If the CSQSNAP DD statement is not in the application JCL or CSQSNAP is not allocated to a data set under TSO, no dump is taken. If this happens, you could include the CSQSNAP DD statement in the application JCL or allocate CSQSNAP to a data set under TSO and rerun the application. However, because some problems are intermittent, it is recommended that you include a CSQSNAP statement in the application JCL or allocate CSQSNAP to a data set in the TSO logon procedure to capture the reason for failure at the time it occurs.

The supplied program CSQBDEFV improves the portability of your application programs. In CSQBDEFV, you can specify the name of a queue manager, or queue sharing group, to be connected to rather than specifying it in the MQCONN or MQCONNX call in an application program. You can create a new version of CSQBDEFV for each queue manager, or queue sharing group. To do this, follow these steps:

1. Copy the IBM MQ assembler program CSQBDEFV from thlqual.SCSQASMS to a user library.
2. The supplied program contains the default subsystem name CSQ1. You can retain this name for testing and installation verification. For production subsystems, you can change the NAME=CSQ1 to your one- to four-character subsystem name, or use CSQ1.

   If you are using queue sharing groups, you can specify a queue sharing group name instead of CSQ1. If you do this, the program issues a connect request to an active queue manager within that group.
3. Assemble and link-edit the program to produce the CSQBDEFV load module. For the assembly, include the library thlqual.SCSQMACS in your SYSLIB concatenation; use the link-edit parameters RENT,AMODE=31,RMODE=ANY. This is shown in the sample JCL in thlqual.SCSQPROC(CSQ4DEFV). Then include the load library in the z/OS Batch or the TSO STEPLIB, ahead of thlqual.SCSQAUTH.

Related concepts

"Set up the operations and control panels" on page 949

To set up the operations and control panels you must first set up the libraries that contain the required panels, EXECs, messages, and tables. To do this, you must take into account which national language feature is to be used for the panels. When you have done this, you can optionally update the main ISPF menu for IBM MQ operations and control panels and change the function key settings.

z/OS *Set up the operations and control panels*

To set up the operations and control panels you must first set up the libraries that contain the required panels, EXECs, messages, and tables. To do this, you must take into account which national language feature is to be used for the panels. When you have done this, you can optionally update the main ISPF menu for IBM MQ operations and control panels and change the function key settings.

- *You need to perform this task once for each z/OS system where you want to run IBM MQ.*
- *You might need to perform this task when migrating from a previous version.*

Setting up the libraries

Follow these steps to set up the IBM MQ operations and control panels:

1. Ensure that all the libraries contained in your concatenations are either in the same format (F, FB, V, VB) and have the same block size, or are in order of decreasing block sizes. Otherwise, you might have problems trying to use these panels.
2. Include the library thlqual.SCSQEXEC in your SYSEXEC or SYSPROC concatenation or activate it using the TSO ALTLIB command. This library, which is allocated with a fixed-block 80 record format during installation, contains the required EXECs.

   It is preferable to put the library into your SYSEXEC concatenation. However, if you want to put it in SYSPROC, the library must have a record length of 80 bytes.

3. Add thlqual.SCSQAUTH and thlqual.SCSQANLx to the TSO logon procedure STEPLIB or activate it using the TSO TSOLIB command, if it is not in the link list or the LPA.
4. You can either add the IBM MQ panel libraries permanently to your ISPF library setup, or allow them to be set up dynamically when the panels are used. For the former choice, you need to do the following:
   a. Include the library containing the operations and control panel definitions in your ISPPLIB concatenation. The name is thlqual.SCSQPNLx, where x is the language letter for your national language.
   b. Include the library containing the required tables in your ISPTLIB concatenation. The name is thlqual.SCSQTBLx, where x is the language letter for your national language.
   c. Include the library containing the required messages in your ISPMLIB concatenation. The name is thlqual.SCSQMSGx, where x is the language letter for your national language.
   d. Include the library containing the required load modules in your ISPLLIB concatenation. The name of this library is thlqual.SCSQAUTH.

   For the latter choice, use the z/OS LIBDEF command. See Examples for a link to various keywords you can use.
5. Test that you can access the IBM MQ panels from the TSO Command Processor panel. This is usually option 6 on the ISPF/PDF Primary Options Menu. The name of the EXEC that you run is CSQOREXX. There are no parameters to specify if you have put the IBM MQ libraries permanently in your ISPF setup as in step 4. If you have not, use the following:

   ```
   CSQOREXX thlqual langletter
   ```

   where `langletter` is a letter identifying the national language to be used:

   C
   Simplified Chinese

   E
   U.S. English (mixed case)

   F
   French

   K
   Japanese

   U
   U.S. English (uppercase)

Updating the ISPF menu

You can update the ISPF main menu to allow access to the IBM MQ operations and control panels from ISPF. The required setting for &ZSEL is:

```
CMD(%CSQOREXX thlqual langletter)
```

For information about `thlqual` and `langletter`, see Step “5” on page 950.

For more details, see the *z/OS: ISPF Dialog Developer's Guide and Reference*.

Updating the function keys and command settings

You can use the normal ISPF procedures for changing the function keys and command settings used by the panels. The application identifier is CSQO.

However, this is not recommended because the help information is not updated to reflect any changes that you have made.

Related concepts

"Include the IBM MQ dump formatting member" on page 951
To be able to format IBM MQ dumps using the Interactive Problem Control System (IPCS), you must update some system libraries.

z/OS *Include the IBM MQ dump formatting member*

To be able to format IBM MQ dumps using the Interactive Problem Control System (IPCS), you must update some system libraries.

- *You need to perform this task once for each z/OS system where you want to run IBM MQ.*
- *You need to perform this task when migrating from a previous version.*

To be able to format IBM MQ dumps using the Interactive Problem Control System (IPCS), copy the data set thlqual.SCSQPROC(CSQ7IPCS) to SYS1.PARMLIB. You should not need to edit this data set.

If you have customized the TSO procedure for IPCS, thlqual.SCSQPROC(CSQ7IPCS) can be copied into any library in the IPCSPARM definition. See *z/OS MVS IPCS User's Guide* for more information.

You must also include the library thlqual.SCSQPNLA in your ISPPLIB concatenation.

To make the dump formatting programs available to your TSO session or IPCS job, you must also include the library thlqual.SCSQAUTH in your STEPLIB concatenation or activate it using the TSO TSOLIB command (even if it is already in the link list or LPA).

Related concepts

"Suppress information messages" on page 951
Your IBM MQ system might produce a large number of information messages. You can prevent selected messages being sent to the console or to the hardcopy log.

z/OS *Suppress information messages*

Your IBM MQ system might produce a large number of information messages. You can prevent selected messages being sent to the console or to the hardcopy log.

- *You need to perform this task once for each z/OS system where you want to run IBM MQ.*
- *You do not need to perform this task when migrating from a previous version.*

If your IBM MQ system is heavily used, with many channels stopping and starting, a large number of information messages are sent to the z/OS console and hardcopy log. The IBM MQ - IMS bridge and buffer manager might also produce a large number of information messages.

If required, you can suppress some of these console messages by using the z/OS message processing facility list, specified by the MPFLSTxx members of SYS1.PARMLIB. The messages you specify still appear on the hardcopy log, but not on the console.

Sample `thlqual.SCSQPROC(CSQ4MPFL)` shows suggested settings for MPFLSTxx. See MPFLSTxx (message processing facility list) for more information.

If you want to suppress selected information messages on the hardcopy log, you can use the z/OS installation exit IEAVMXIT. You can set the following bit switches ON for the required messages:

CTXTRDTM
Delete the message.

The message is not displayed on consoles or logged in hardcopy.

CTXTESJL
Suppress from job log.

The message does not go into the JES job log.

CTXTNWTP
Do not carry out WTP processing.

The message is not sent to a TSO terminal or to the system message data set of a batch job.

Note:

1. For full details on the other parameters, see MVS Installation Exits.
2. You are not recommended to suppress messages other than those in the suggested suppression list, CSQ4MPFL.

In addition you can specify the extra parameter:

EXCLMSG
Specifies a list of messages to be excluded from any log.

Messages in this list are not sent to the z/OS console and hardcopy log. See EXCLMSG in "Utilización de CSQ6SYSP" on page 925 for further information.

Related tasks

"Testing a queue manager on z/OS" on page 967
When you have customized or migrated your queue manager, you can test it by running the installation verification programs and some of the sample applications shipped with IBM MQ for z/OS.

z/OS Configuring the queue sharing group

If you want to use shared queues for high availability, use these topics as a step by step guide for configuring the queue sharing group.

When you have completed the steps in this part of the process for setting up your IBM MQ for z/OS system, you should "Tailor your system parameter module" on page 923 to add queue sharing group data. You need to modify CSQ6SYSP to specify the QSGDATA parameter.

z/OS *Set up the Db2 environment*

If you are using queue sharing groups you must create the required Db2 objects by customizing and running a number of sample jobs.

Set up the Db2 environment

You must create and bind the required Db2 objects by customizing and running a number of sample jobs.

- Repeat this task for each Db2 data-sharing group.
- You need to perform the `bind` and `grant` steps when migrating from a previous version.
- Omit this task if you are not using queue sharing groups.

  If you later want to use queue sharing groups, perform this task at that time.

IBM MQ provides two equivalent sets of jobs. Those with the CSQ45 prefix are for compatibility with earlier versions of IBM MQ and for use with IBM MQ version 11 and earlier. If you are setting up a new data-sharing group with Db2 V12 or later, you are encouraged to use the jobs with CSQ4X prefix, as these jobs exploit more recent Db2 capabilities for dynamic sizing and Universal Table Spaces (UTS).

The following steps must be performed for each new Db2 data-sharing group. All the sample JCL is in thlqual.SCSQPROC.

1. Customize and execute sample JCL CSQ4XCSG to create the storage group that is to be used for the IBM MQ database, table spaces, and tables.
2. Customize and execute sample JCL CSQ4XCDB to create the database to be used by all queue managers that are connecting to this Db2 data-sharing group.
3. Customize and execute sample JCL CSQ4XCTS to create the table spaces that contain the queue manager and channel initiator tables used for queue sharing groups.

4. Customize and execute sample JCL CSQ4XCTB to create the 15 Db2 tables and associated indexes. Do not change any of the row names or attributes.
5. Customize and execute sample JCL CSQ45BPL to bind the Db2 plans for the queue manager, utilities, and channel initiator.
6. Customize and execute sample JCL CSQ45GEX to grant execute authority to the plans for the user IDs that are used by the queue manager, utilities, and channel initiator. The user IDs for the queue manager and channel initiator are the user IDs under which their started task procedures run. The user IDs for the utilities are the user IDs under which the batch jobs can be submitted.

   The names of the appropriate plans are shown in the following table.

User	**Plans (LTS)**	**Plans (CD)**
Queue manager	CSQ5A 930, CSQ5C 930, CSQ5D 930, CSQ5K 930, CSQ5L 930, CSQ5M 930, CSQ5P 930, CSQ5R 930, CSQ5S 930, CSQ5T 930, CSQ5U 930, CSQ5W 930	CSQ5A 9X0, CSQ5C 9X0, CSQ5D 9X0, CSQ5K 9X0, CSQ5L 9X0, CSQ5M 9X0, CSQ5P 9X0, CSQ5R 9X0, CSQ5S 9X0, CSQ5T 9X0, CSQ5U 9X0, CSQ5W 9X0
SDEFS function of the CSQUTIL batch utility	CSQ52 930	CSQ52 9X0
CSQ5PQSG and CSQJUCNV batch utilities	CSQ5B 930	CSQ5B 9X0
CSQUZAP service utility	CSQ5Z 930	CSQ5Z 9X0

In the event of a failure during Db2 setup, the following jobs can be customized and executed:

- CSQ45DTB to drop the tables and indexes.
- CSQ4XDTS to drop the table spaces.
- CSQ4XDDB to drop the database.
- CSQ4XDSG to drop the storage group.

Note: If these jobs fail because of a Db2 locking problem it is probably due to contention for a Db2 resource, especially if the system is being heavily used. Resubmit the jobs later. It is preferable to run these jobs when the system is lightly used or quiesced.

See Db2 Administration in *Db2 for z/OS 12.0.0* for more information about setting up Db2.

See Planificación en z/OS for information about Db2 table sizes.

Related concepts

“Set up the coupling facility” on page 954

If you are using queue sharing groups, define the coupling facility structures used by the queue managers in the queue sharing group (QSG) in the coupling facility Resource Management (CFRM) policy data set, using IXCMIAPU.

z/OS Set up the coupling facility

If you are using queue sharing groups, define the coupling facility structures used by the queue managers in the queue sharing group (QSG) in the coupling facility Resource Management (CFRM) policy data set, using IXCMIAPU.

See Administrative data utility for more information on IXCMIAPU.

- Repeat this task for each queue sharing group.
- You might need to perform this task when migrating from a previous version.
- Omit this task if you are not using queue sharing groups.

  If you later want to use queue sharing groups, perform this task at that time.

All the structures for the queue sharing group start with the name of the queue sharing group. Define the following structures:

- An administrative structure called *qsg-name* CSQ_ADMIN. This structure is used by IBM MQ itself and does not contain any user data.
- A system application structure called *qsg-name* CSQSYSAPPL. This structure is used by IBM MQ system queues to store state information.
- One or more structures used to hold messages for shared queues. These can have any name you choose up to 16 characters long.
  - The first four characters must be the queue sharing group name. (If the queue sharing group name is less than four characters long, it must be padded to four characters with @ symbols.)
  - The fifth character must be alphabetic and subsequent characters can be alphabetic or numeric. This part of the name (without the queue sharing group name) is what you specify for the CFSTRUCT name when you define a shared queue, or a CF structure object.

  You can use only alphabetic and numeric characters in the names of structures used to hold messages for shared queues, you cannot use any other characters (for example, the _ character, which is used in the name of the administrative structure).

Sample control statements for IXCMIAPU are in data set thlqual.SCSQPROC(CSQ4CFRM). Customize these and add them to your IXCMIAPU job for the coupling facility and run it.

When you have defined your structures successfully, activate the CFRM policy that is being used. To do this, issue the following z/OS command:

```
SETXCF START,POLICY,TYPE=CFRM,POLNAME= policy-name
```

For information about planning CF structures and their sizes, see Defining coupling facility resources.

Related concepts

"Implement your ESM security controls" on page 915
Implement security controls for queue managers and the channel initiator.

z/OS Set up the SMDS environment

If you want to use SMDS to offload messages on shared queues, set up the SMDS offload storage environment.

- *Perform this task for each queue manager and structure in the queue sharing group that you want to configure to offload data to SMDS.*
- *If you want to configure additional structures to offload data to SMDS later, this task can be performed again at that time.*
- *Omit this task if you are not using queue sharing groups.*

  If you later want to use queue sharing groups, perform this task at that time.

Set up the SMDS environment

1. Estimate structure and data set space requirements. See Shared message data set capacity considerations.
2. Allocate and preformat data sets. See Creating a shared message data set.
3. When you define the CF structure to IBM MQ, ensure that you define the CFSTRUCT with CFLEVEL(5) and OFFLOAD(SMDS).

Related concepts

"Set up the coupling facility" on page 954
If you are using queue sharing groups, define the coupling facility structures used by the queue managers in the queue sharing group (QSG) in the coupling facility Resource Management (CFRM) policy data set, using IXCMIAPU.

z/OS *Add the IBM MQ entries to the Db2 tables*

If you are using queue sharing groups, run the CSQ5PQSG utility to add queue sharing group and queue manager entries to the IBM MQ tables in the Db2 data-sharing group.

- *Repeat this task for each IBM MQ queue sharing group and each queue manager.*
- *You might need to perform this task when migrating from a previous version.*
- *Omit this task if you are not using queue sharing groups.*

  If you later want to use queue sharing groups, perform this task at that time.

Run CSQ5PQSG for each queue sharing group and each queue manager that is to be a member of a queue sharing group.

Perform the following actions in the specified order:

1. Add a queue sharing group entry into the IBM MQ Db2 tables using the ADD QSG function of the CSQ5PQSG program. A sample is provided in thlqual.SCSQPROC(CSQ45AQS).

   Perform this function once for each queue sharing group that is defined in the Db2 data-sharing group. The queue sharing group entry must exist before adding any queue manager entries that reference the queue sharing group.
2. Add a queue manager entry into the IBM MQ Db2 tables using the ADD QMGR function of the CSQ5PQSG program. A sample is provided in thlqual.SCSQPROC(CSQ45AQM).

   Perform this function for each queue manager that is to be a member of the queue sharing group.

   Note:

   a. A queue manager can only be a member of one queue sharing group.

   b. You must have RRS running to be able to use queue sharing groups.

Related concepts

"Tailor your system parameter module" on page 923
The IBM MQ system parameter module controls the logging, archiving, tracing, and connection environments that IBM MQ uses in its operation. A default module is supplied. You should create your own system parameter module as some parameters, for example data set names, are usually site specific.

z/OS *Implement ESM security controls for the queue sharing group*

Implement security controls for all queue managers in a queue sharing group, to access Db2 and the coupling facility list structures.

- *Repeat this task for each IBM MQ queue manager in a queue sharing group.*
- *You might need to perform this task when migrating from a previous version.*

Ensure that the user IDs associated with the queue manager, channel initiator, and the utilities have authority to establish an RRSAF connection to each Db2 subsystem with which you want to establish a

connection. The user IDs for the queue manager and channel initiator are the user IDs under which their started task procedures run.

The user IDs for the utilities are the user IDs under which the batch jobs can be submitted. The RACF profile to which the user ID requires READ access is `Db2ssid.RRSAF` in the DSNR resource class

The user IDs associated with each queue manager in a queue sharing group need to be granted the appropriate level of access to the coupling facility list structures. The RACF class is FACILITY.

The following user IDs require ALTER access:

- The queue manager ID to the `IXLSTR.structure-name` profile
- The user ID running CSQ5PQSG

Related concepts

"Implement your ESM security controls" on page 915
Implement security controls for queue managers and the channel initiator.

z/OS Configuring Advanced Message Security for z/OS

Use these topics as a step by step guide for configuring Advanced Message Security (AMS).

Before you begin

Before you start to configure AMS, ensure that the following queue manager configuration steps have been performed:

1. Add the CSQ0DRTM module to the LPA, as described in "Actualizar LPA y la lista de enlaces de z/OS" on page 904.
2. Add an entry for CSQ0DSRV to the z/OS program properties table (PPT), as described in "Update the z/OS program properties table" on page 908.
3. Include the CSQ4INSM member in the CSQINP2 concatenation of queue manager started task procedure, as described in "Customize the initialization input data sets" on page 916.
4. Enable AMS using the AMSPROD attribute. See product usage recording with IBM MQ for z/OS products for more details.

What to do next

Configure policies for queues protected by AMS. Security policies are described in Administering Advanced Message Security security policies.

There are examples of AMS configurations in Example configurations on z/OS.

z/OS *Create procedures for Advanced Message Security*

Each IBM MQ subsystem that is to be configured to use Advanced Message Security (AMS) requires a cataloged procedure to start the AMS address space. You can create your own or use the IBM-supplied procedure library.

Procedure

1. Copy the sample started task procedure *thlqual*.SCSQPROC(CSQ4AMSM) to your SYS1.PROCLIB or, if you are not using SYS1.PROCLIB, your procedure library. Name the procedure xxxxAMSM, where xxxx is the name of your IBM MQ subsystem. For example, CSQ1AMSM would be the AMS started task procedure for queue manager CSQ1.
2. Make a copy for each IBM MQ subsystem that you are going to use.
3. Tailor the procedures to your requirements using the instructions in the sample procedure CSQ4AMSM. You can also use symbolic parameters in the JCL to allow the procedure to be modified when it is started.

4. Review and optionally change the parameters passed to the AMS task using the Language Environment® _CEE_ENVFILE file. The sample thlqual.SCSQPROC(CSQ40ENV) lists the supported parameters.
5. Repeat steps 1 to 4 for each IBM MQ queue manager.

What to do next

"Set up the Advanced Message Security started task user ID" on page 957

z/OS Set up the Advanced Message Security started task user ID

The Advanced Message Security (AMS) task requires a user ID that allows it to be known as a z/OS UNIX System Services (z/OS UNIX) process.

About this task

In addition, the users that the task works on behalf of must also have an appropriate definition of a UNIX UID (user ID) and GID (group ID) so these users are known as z/OS UNIX System Services users. For more information on defining z/OS UNIX System Services UIDs and GIDs, see *z/OS: Security Server RACF Security Administrator's Guide*.

Review *z/OS UNIX System Services Planning* to ensure that you understand the security differences between traditional UNIX security and z/OS UNIX security. This allows you to administer the Advanced Message Security task according to your installation's security policy for deploying and running privileged z/OS UNIX System Services processes.

The primary difference between traditional UNIX security and z/OS security is that the Kernel services support two levels of appropriate privileges: UNIX level and z/OS UNIX level.

Depending on your installation's security policy, the Advanced Message Security task can either run with superuser authority (uid(0)), or with its RACF identity permitted to the RACF FACILITY class BPX.DAEMON and BPX.SERVER profiles, as this task must be able to assume the RACF identity of its users.

If the latter method is used, or you have already activated the BPX.DAEMON or BPX.SERVER profiles, the Advanced Message Security task program (thlqual.SCSQAUTH(CSQ0DSRV)) must be located in RACF program-controlled libraries.

Note: Choose the user ID for this task carefully because the Advanced Message Security recipient certificates are loaded into a key ring associated with this user ID. This consideration is discussed in Using certificates on z/OS .

The steps shown here describe how to set up the Advanced Message Security started task user. The steps use RACF commands as examples. If you are using a different security manager, you should use equivalent commands.

Note: The examples in this section assume that you have activated generic profile command processing for the RACF STARTED, FACILITY, and SURROGAT classes and generic profile checking. For more information on how RACF handles generic profiles, see *z/OS: Security Server RACF Command Language Reference*.

Procedure

1. Define the Advanced Message Security started task user to RACF. The examples in this section use the user ID WMQAMSM.

   ```
   ADDUSER WMQAMSM NAME('AMS user') OMVS (UID(0)) DFLTGRP(group)
   ```

   Select a default 'group' as appropriate to your installation standards.

Note: If you do not want to grant z/OS UNIX superuser authority (UID(0)), then you must permit the Advanced Message Security user ID to the BPX.DAEMON and BPX.SERVER facility class profiles:

```
PERMIT BPX.DAEMON CLASS(FACILITY) ID(WMQAMSM) ACCESS(READ)
```

and the Advanced Message Security task program (*thlqual*.SCSQAUTH(CSQ0DSRV)) must be located in a RACF program-controlled library.

To make your SCSQAUTH library program controlled, you can use the following command:

```
RALTER PROGRAM * ADDMEM('thlqual.SCSQAUTH'//NOPADCHK) -or-
RALTER PROGRAM ** ADDMEM('thlqual.SCSQAUTH'//NOPADCHK)
SETROPTS WHEN(PROGRAM) REFRESH
```

You must also enable program control for the national language library (*thlqual*.SCSQANLx) that is used by the Advanced Message Security task.

2. Determine if the RACF STARTED class is active. If it is not, activate the RACF STARTED class:

```
SETROPTS CLASSACT(STARTED)
```

3. Define a started class profile for the Advanced Message Security tasks, specifying the user ID you selected or created in step 1:

```
RDEFINE STARTED qmgrAMSM.* STDATA(USER(WMQAMSM))
```

where *qmgr* is the prefix of the started task name. For example, the started task may be named CSQ1AMSM. In this case, you would substitute *qmgr*AMSM.* with CSQ1AMSM.*.

The AMS started tasks must be named *qmgr*AMSM.

4. Use the **SETROPTS** RACF command to refresh the in-storage RACLISTed STARTED class profiles:

```
SETROPTS RACLIST(STARTED) REFRESH
```

5. The Advanced Message Security task temporarily assumes the identity of the host user ID of the requestor during protection processing of IBM MQ messages. Therefore, it is necessary to define profiles in the SURROGAT class for each user ID that can make requests.

   If the RACF SURROGAT class is active, defining a single generic profile allows the Advanced Message Security task to assume the identity of any user. The check is ignored if the SURROGAT class is not active. The SURROGAT profiles needed are described in *z/OS UNIX System Services Planning*.

   To define profiles in the SURROGAT class:

   a) Activate the RACF SURROGAT class using the RACF SETROPTS command:

```
SETROPTS CLASSACT(SURROGAT)
```

   b) Activate generic profile processing for the RACF SURROGAT class:

```
SETROPTS GENERIC(SURROGAT)
```

   c) Activate generic profile command processing for the RACF SURROGAT class:

```
SETROPTS GENCMD(SURROGAT)
```

   d) Define a generic profile in the SURROGAT class:

```
RDEFINE SURROGAT BPX.SRV.* UACC(NONE)
```

e) Permit the Advanced Message Security user ID to the generic SURROGAT class profile:

```
PERMIT BPX.SRV.* CLASS(SURROGAT) ID(WMQAMSM) ACCESS(READ)
```

Note: You can define more specific profiles if you want to restrict specific users to be processed by the Advanced Message Security task, as described in *z/OS UNIX System Services Planning*.

For example, a profile called BPX.SRV.MQUSER1 controls whether the AMS task can assume the identity of the user ID MQUSER1.

f) Permit the Advanced Message Security user ID to the BPX.SERVER facility (if not already done in Creating the certificates and key rings):

```
PERMIT BPX.SERVER CLASS(FACILITY) ID(WMQAMSM) ACCESS(READ)
```

g) Use the **SETROPTS** RACF command to refresh the in-storage RACLISTed started class profiles:

```
SETROPTS RACLIST(SURROGAT) REFRESH
SETROPTS RACLIST(FACILITY) REFRESH
```

6. The Advanced Message Security task uses the facilities provided by z/OS System SSL services to open SAF-managed key rings. The underlying System Authorization Facility (SAF) that accesses the contents of the key rings is controlled by RACF, or an equivalent security manager.

   This service is the IRRSDL00 (R_datalib) callable service. This callable service is protected with the same profiles used to protect the RACF RACDCERT commands that are defined to the RACF FACILITY class. Thus, the Advanced Message Security user ID must be permitted to the profiles using these commands:

   a) If you have not already done so, define a RACF generic profile to the RACF FACILITY class that protects the RACDCERT command and the IRRSDL00 callable service:

   ```
   RDEFINE FACILITY IRR.DIGTCERT.* UACC(NONE)
   SETROPTS RACLIST(FACILITY) REFRESH
   ```

   b) Grant authority to the started task user ID to the RACF generic profile:

   ```
   PERMIT IRR.DIGTCERT.* CLASS(FACILITY) ID(WMQAMSM) ACC(READ)
   ```

   Alternatively, you can grant READ access to the data service task user's keyring in the RDATALIB class as follows:

   ```
   PERMIT WMQASMD.DRQ.AMS.KEYRING.LST CLASS(RDATALIB) ID(WMQAMSM) ACC(READ)
   ```

7. Configure resource security:

   a) The Advanced Message Security started task user requires authority to connect to the queue manager as a batch application.

   If your queue manager has connection security enabled, grant the AMS task authority to connect to the queue manager with this command:

   ```
   PERMIT hlq.BATCH CLASS(MQCONN) ID(WMQAMSM) ACC(READ)
   ```

   where *hlq* can be either the queue manager name queue sharing group name.

   For further information, see Connection security profiles for batch connections.

   b) The Advanced Message Security started task user requires authority to browse the SYSTEM.PROTECTION.POLICY.QUEUE.

If queue security is active on the queue manager, grant the AMS user authority to access the queue with these commands:

```
RDEFINE MQQUEUE hlq.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
PERMIT hlq.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE) ID(WMQAMSM) ACCESS(READ)
```

where *hlq* can be either the queue manager name queue sharing group name.

If the queue manager is using mixed case profiles, define the profile in the MXQUEUE class instead.

To manage AMS security policies using the CSQ0UTIL utility, administrators need access to put messages to the SYSTEM.PROTECTION.POLICY.QUEUE. This is performed by granting UPDATE access to the profile protecting the queue.

For further information, see Profiles for queue security.

What to do next

"Grant RACDCERT permissions to the security administrator for Advanced Message Security" on page 960

z/OS *Grant RACDCERT permissions to the security administrator for Advanced Message Security*

Your Advanced Message Security security administrator requires authority to use the RACDCERT command to create and manage digital certificates.

Procedure

- Identify the appropriate user ID for this role and grant permission to use the RACDCERT command. For example:

  ```
  PERMIT IRR.DIGTCERT.* CLASS(FACILITY) ID(admin) ACCESS(CONTROL)
  SETROPTS RACLIST(FACILITY) REFRESH
  ```

  where `admin` is the user ID of your Advanced Message Security security administrator.

What to do next

"Grant users resource permissions for Advanced Message Security" on page 960

z/OS *Grant users resource permissions for Advanced Message Security*

Advanced Message Security users require relevant resource permissions.

About this task

Advanced Message Security users, that is users that are putting or getting Advanced Message Security protected messages, require:

- An OMVS segment associated with their user id
- Permissions for IRR.DIGTCERT.LISTRING or RDATALIB
- Permissions for ICSF class CSFSERV and CSFKEYS profiles
- Permission to put to the SYSTEM.PROTECTION.ERROR.QUEUE

The Advanced Message Security task temporarily assumes the identity of its clients; that is, the task acts as a surrogate of the z/OS user ID of users of Advanced Message Security during the processing of IBM MQ messages to queues that are protected by Advanced Message Security.

In order for the task to assume the z/OS identity of a user, the client z/OS user ID must have a defined OMVS segment associated with its user profile.

As an administration aid, RACF provides the ability to define a default OMVS segment that may be associated with RACF user and group profiles. This default is used if the z/OS user ID or group profile does not have an OMVS segment explicitly defined. If you plan to have a large number of users using Advanced Message Security, you might choose to use this default rather than explicitly defining the OMVS segment for each user.

The *z/OS: Security Server RACF Security Administrator's Guide* contains the detailed procedure for defining default OMVS segments. Review the procedure as outlined in this publication to determine if the definition of default OMVS segments in RACF User and Group profiles is appropriate to your installation.

Procedure

1. Grant READ permission to the IRR.DIGTCERT.LISTRING profile in the FACILITY class:
   - To grant READ permission to the IRR.DIGTCERT.LISTRING profile in the FACILITY class to all users, issue this command:

     ```
     RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(READ)
     ```

   - To grant READ permission to the IRR.DIGTCERT.LISTRING profile in the FACILITY class on a per user basis, issue this command:

     ```
     PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID(userid) ACCESS(READ)
     ```

     where userid is the name of the Advanced Message Security user.
   - Alternatively, use the RDATALIB class to grant access to specific key rings. The RDATALIB permissions take precedence over IRR.DIGTCERT.LISTRING permissions. For example:

     ```
     PERMIT user.DRQ.AMS.KEYRING.LST CLASS(RDATALIB) ID(user) ACC(READ)
     ```

2. If you are using ICSF-managed certificates and private keys, Advanced Message Security users require access to certain class CSFSERV and CSFKEYS profiles. This access is detailed in the following table:

Table 64. Required user access to class CSFSERV and CSFKEYS profiles

Class	Profile	Permission
CSFSERV	CSFDSG	READ
CSFSERV	CSFPKE	READ
CSFSERV	CSFPKD	READ
CSFSERV	CSFDSV	READ
CSFKEYS	ICSF PKDS Label	READ

3. Applications that perform operations on queues with AMS policies defined need access to put messages to SYSTEM.PROTECTION.ERROR.QUEUE. Grant put access to the queue with these commands:

   ```
   RDEFINE MQQUEUE hlq.SYSTEM.PROTECTION.ERROR.QUEUE UACC(NONE)
   PERMIT hlq.SYSTEM.PROTECTION.ERROR.QUEUE CLASS(MQQUEUE) ID(userId) ACCESS(UPDATE)
   ```

   where *hlq* can be either the queue manager name queue sharing group name, and *userID* is the application user ID.

What to do next

"Create key rings for Advanced Message Security" on page 962

z/OS *Create key rings for Advanced Message Security*

Certificates used by Advanced Message Security (AMS) for signing and encryption are stored in z/OS SAF key rings. You need to create these key rings and certificates before you can use AMS.

About this task

Advanced Message Security accesses certificates in the following key rings:

- A single key ring owned by the AMS address space user.
- Key rings owned by the individual users that send or receive messages on queues with AMS policies defined.

These key rings must all be named `drq.ams.keyring`.

There is more information on key rings and certificates used by AMS, and an example scenario, in Using certificates on z/OS.

Follow these steps to create the key rings required by AMS, and connect certificates to the key rings. You must create the key ring owned by the AMS address space user before starting AMS. You can create the keys rings owned by the users that send or receive messages at any time.

Procedure

1. Issue the following command to create a key ring owned by the AMS address space user:

   ```
   RACDCERT ID(amsUser) ADDRING(drq.ams.keyring)
   ```

   where *amsUser* is the user ID of the AMS address space.
2. Create a key ring for each user that sends or receives messages protected by AMS by issuing the command in step 1 for each user ID.
3. Connect the certificate authority (CA) certificate for the issuer of the user certificates to the key ring owned by the AMS address space user ID. Issue the following command:

   ```
   RACDCERT ID(amsUser) CONNECT(CERTAUTH LABEL('caLabel') RING(drq.ams.keyring))
   ```

   where *amsUser* is the user ID of the AMS address space, and *caLabel* is the label of the CA certificate.

   If you are using RACF as your CA, and need to create a certificate authority certificate, follow the example in Defining a local Certificate Authority certificate.
4. If you are using privacy or confidentiality security policies to encrypt messages on queues protected by AMS, connect the certificates of message recipients to the key ring owned by the AMS address space user ID. Issue the following command:

   ```
   RACDCERT ID(amsUser) CONNECT(ID(userId) LABEL('certLabel')
   RING(drq.ams.keyring) USAGE(SITE))
   ```

   where *amsUser* is the user ID of the AMS address space, *userId* is the message recipient, and *certLabel* is the label of the user's certificate.

   The `USAGE(SITE)` attribute prevents the private key from being accessible in the key ring.

   If you are creating your own certificates with RACF, follow the example in Creating a digital certificate with a private key to create the certificate.
5. Connect the certificates of each user that sends or receives messages protected by AMS to a key ring owned by the user. The certificate must be connected as the default certificate in the key ring. Issue the following command:

   ```
   RACDCERT ID(userId) CONNECT(ID(userId) LABEL('certLabel')
   RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
   ```

   where *userId* is the user that is sending or receiving messages, and *certLabel* is the label of the user's certificate.

Notes:

a. Steps "2" on page 962 and "5" on page 962 are not required if the application opens a queue only for output, and sends messages to queues protected by an AMS confidentiality policy.

b. Steps "2" on page 962 and "5" on page 962 are not required if the application opens a queue only for input/browse, and receives messages from queues protected by an AMS integrity policy.

What to do next

"Enable Advanced Message Security" on page 963

z/OS *Enable Advanced Message Security*

Security policy capability for a queue manager is controlled by the SPLCAP parameter in the system parameter module.

About this task

Follow these steps to enable Advanced Message Security (AMS) for a single queue manager.

This task requires you to make a change to the system parameter module. See "Tailor your system parameter module" on page 923 for more information on creating and customizing the system parameter module.

Procedure

1. Set **SPLCAP** to YES in CSQ6SYSP. See "Utilización de CSQ6SYSP" on page 925 for more information on the CSQ6SYSP macro.
2. Set the **AMSPROD** to either AMS, ADVANCED, or ADVANCEDVUE depending on your licence entitlement. See using CSQ6USGP for more information on the CSQ6USGP macro.
3. Recompile the system parameter module.
4. Restart the queue manager with the updated system parameter module. The AMS address space is started automatically when the queue manager starts.

z/OS Configuring the mqweb server

Use these topics as a step by step guide for configuring the mqweb server.

Related tasks

"Configuración de IBM MQ Console y REST API" on page 850

El servidor mqweb que aloja IBM MQ Console y REST API se proporciona con una configuración predeterminada. Para utilizar cualquiera de estos componentes es necesario realizar una serie de tareas de configuración, como la configuración de seguridad para permitir a los usuarios iniciar la sesión. Este tema describe todas las opciones de configuración que están disponibles.

z/OS *Creación del servidor mqweb*

Si ha instalado los componentes web de IBM MQ for z/OS UNIX System Services Web Components y desea utilizar la IBM MQ Console o la REST API, tendrá que crear y personalizar el servidor mqweb.

Antes de empezar

Antes de ejecutar el script **crtmqweb** para crear el servidor mqweb, establezca la variable de entorno JAVA_HOME para que haga referencia a una versión de 64 bits de Java en el sistema.

El IBM MQ Console y el administrative REST API requieren el SYSTEM.REST.REPLY.QUEUE que se va a crear. Cree esta cola utilizando el ejemplo **CSQ4INSG** en "Customize the initialization input data sets" en la página 916.

Atención: Al iniciar el servidor mqweb, si encuentra el mensaje de error CWWKG0014E, tal como se muestra en la salida siguiente:

```
     Launching mqweb (MQM MVS/ESA V9 R2.0/wlp...) (en_US)
               ÝAUDIT   ¨ CWWKE0001I: The server mqweb has been
launched.
               ÝWARNING ¨ CWWKF0009W: The server has not been configured to install any
features.
               ÝAUDIT   ¨ CWWKF0011I: The mqweb server is ready to run a smarter planet.
The mqweb server started in 6.348 seconds.
               ÝERROR   ¨ CWWKG0014E: The configuration parser detected an XML syntax
error while parsing the root of the configuration and the referenced configuration
documents.
                                Error: An invalid XML character (Unicode: 0x4c) was found
in the prolog of the document.
                                 File: file:<your filepath>/servers/mqweb/server.xml Line:
1 Column: 1
```

debe comprobar el valor z/OS de AUTOCVT (convertir automáticamente archivos de un conjunto de códigos a otro) y ajustar el valor según sea necesario realizando una de las acciones siguientes.

En un terminal USS:
Emita el mandato: echo **$_BPXK_AUTOCVT** para visualizar el valor de esta variable de entorno. Si la variable de entorno no está definida, no se visualiza ningún valor.

Para establecer la variable de entorno, consulte _BPXK variables de entorno.

En todo el sistema:
El ejemplo 6 en Visualización del estado de z/OS UNIX System Services (OMVS) muestra cómo visualizar el valor de la sentencia AUTOCVT de todo el sistema en BPXPRMxx.

Para establecer la variable de entorno en todo el sistema, utilice la sentencia AUTOCVT en BPXPRMxx.

Si la variable de entorno _BPXX_AUTOCVT se establece en un terminal USS, altera temporalmente el valor de todo el sistema de la sentencia AUTOCVT en BPXPRMxx.

Acerca de esta tarea

- Complete esta tarea una vez para cada sistema z/OS en el que desee ejecutar IBM MQ Console o REST API.
- Para utilizar administrative REST API, necesita un servidor mqweb para cada versión de IBM MQ que se esté ejecutando. Por ejemplo, si está ejecutando IBM MQ 9.4.0, 9.3.5 y 9.3.0, necesita tres servidores mqweb diferentes.
- Es posible que tenga que renovar o modificar la configuración del servidor al migrar desde una versión anterior.

IBM MQ Console y REST API requieren la creación de un único servidor WebSphere Liberty, denominado mqweb.

Los archivos de registro y configuración de servidor se almacenan todos en el directorio de usuarios de Liberty.

El servidor mqweb debe configurarse con un ID de producto (PID) bajo el que se ejecuta. El PID se establece cuando se crea el servidor mqweb. Utilice el mismo PID que se utiliza para ejecutar los gestores de colas locales a los que se conecta el servidor mqweb.

Nota: si los gestores de colas locales se ejecutan con varios PID diferentes, elija uno de ellos para ejecutar el servidor mqweb.

Para obtener más información sobre los PID y cómo se utilizan en z/OS, consulte Registro de uso del producto con productos IBM MQ for z/OS.

Es posible cambiar el PID bajo el que se ejecuta el servidor mqweb, después de su creación, utilizando el setmqweb pid dominio.

Realice los pasos siguientes para crear el servidor mqweb:

Procedimiento

1. Decida con qué PID se ejecuta el servidor mqweb.
2. Elija una ubicación adecuada para el directorio de usuarios de Liberty.

   El ID de usuario, bajo el cual se ejecuta el servidor mqweb, necesita acceso de lectura y escritura a este directorio de usuarios y su contenido. Como este directorio de usuario contiene archivos de registro, además de la configuración del servidor, cree este directorio en un sistema de archivos independiente.

   Nota: Existe una cantidad significativa de E/S de disco cuando se inicia el servidor mqweb. Para reducir el tiempo que se tarda en iniciar el servidor mqweb, asegúrese de que tanto el sistema de archivos de IBM MQ instalación z/OS UNIX como el sistema de archivos de directorio de usuario de Liberty tienen en cuenta sysplex-ico o están montados localmente en el sistema donde se ejecuta el servidor mqweb.
3. En z/OS UNIX System Services, cambie el directorio de trabajo actual a `PathPrefix/web/bin` emitiendo el mandato siguiente:

   ```
   cd PathPrefix/web/bin
   ```

   donde *PathPrefix* es la vía de acceso de instalación de IBM MQ for z/OS UNIX System Services Components .
4. Cree el directorio de usuario Liberty que contiene la definición de servidor mqweb de plantilla, ejecutando el script **crtmqweb** .

   El formato del mandato **crtmqweb** es:

   ```
   crtmqweb user_directory -p pid_value
   ```

   donde:

   directorio_usuario
   Es el directorio de usuarios de Liberty decidido en el paso “2” en la página 965. Este parámetro es opcional. Si no se especifica este parámetro, se utiliza un directorio de usuario Liberty predeterminado de `/var/mqm/web/installation1` .

   valor_pid
   Indica el PID bajo el que se ejecuta el servidor mqweb. Este PID es el que ha elegido en el paso “1” en la página 965. *valor_pise* es uno de los valores siguientes:

   MQ
   El servidor mqweb se ejecuta bajo PID IBM MQ for z/OS (5655-MQ9).

   VUE
   El servidor mqweb se ejecuta bajo PID IBM MQ for z/OS Value Unit Edition (VUE) (5655-VU9).

   ADVANCEDVUE
   El servidor mqweb se ejecuta bajo el PID IBM MQ Advanced for z/OS VUE (5655-AV1),

   Por ejemplo, para crear el servidor mqweb con un directorio de usuario Liberty de `/usr/mqweb` y un PID de IBM MQ Advanced for z/OS VUE (5655-AV1), ejecute el mandato siguiente:

   ```
   ./crtmqweb /usr/mqweb -p ADVANCEDVUE
   ```
5. Cambie la propiedad de los directorios y archivos en el directorio de usuario Liberty , para que pertenezcan al ID de usuario y al grupo bajo el que se ejecuta el servidor mqweb, utilizando el mandato:

   ```
   chown -R userid:group path
   ```

   Para otorgar al grupo el acceso de escritura a la vía de acceso, emita el mandato:

   ```
   chmod -R 770 path
   ```

Qué hacer a continuación

“Creating a procedure for the mqweb server” en la página 966

Tareas relacionadas

“Configuración de IBM MQ Console y REST API” en la página 850

El servidor mqweb que aloja IBM MQ Console y REST API se proporciona con una configuración predeterminada. Para utilizar cualquiera de estos componentes es necesario realizar una serie de tareas de configuración, como la configuración de seguridad para permitir a los usuarios iniciar la sesión. Este tema describe todas las opciones de configuración que están disponibles.

z/OS *Creating a procedure for the mqweb server*

If you installed the IBM MQ for z/OS UNIX System Services Web Components, and want to use the IBM MQ Console, or the REST API, you need to create a cataloged procedure to start the mqweb server. The mqweb server is a Liberty server that hosts the IBM MQ Console and the REST API.

- You need to perform this task once for each z/OS system where you want to run the IBM MQ Console or REST API.
- You need a mqweb server for each version of IBM MQ that is running. For example, a started task called MQWB0910 for queue managers at IBM MQ for z/OS 9.1.0 and a started task called MQWB0905 for queue managers at IBM MQ for z/OS 9.0.5.

  If you have only one queue manager on the z/OS system, you can run a single Liberty server started task, and change the libraries it uses when you migrate your queue manager.
- You might need to modify the cataloged procedure when migrating from a previous version.

Carry out the following procedure to create a cataloged procedure:

1. Copy the sample started task procedure `thlqual.SCSQPROC(CSQ4WEBS)` to your procedure library.

   Name the procedure according to the standards of your enterprise.

   For example `MQWB0910`, indicating that this is the cataloged procedure for the IBM MQ for z/OS 9.1.0 mqweb server.
2. Tailor the procedure to your requirements using the instructions in the sample procedure CSQ4WEBS.

   Note that the Liberty user directory is the directory specified when the **crtmqweb** script was run to create the mqweb server definition.

   See “Creación del servidor mqweb” on page 963 for details.

   Note: Ensure that you specify **Caps off** when you edit the member, as the file has lowercase data.
3. Authorize the procedure to run under your external security manager.
4. Use IBM Workload Manager (WLM) to classify this address space.

   The mqweb server is an IBM MQ application, and users interact with this application. The application does not need to be high importance in WLM, and a service class of **STCUSER** might be suitable.

What to do next

Follow the steps in “Configuración básica para el servidor mqweb” on page 850 to finish configuring the mqweb server.

Related tasks

“Configuración de IBM MQ Console y REST API” on page 850

El servidor mqweb que aloja IBM MQ Console y REST API se proporciona con una configuración predeterminada. Para utilizar cualquiera de estos componentes es necesario realizar una serie de tareas de configuración, como la configuración de seguridad para permitir a los usuarios iniciar la sesión. Este tema describe todas las opciones de configuración que están disponibles.

z/OS Testing a queue manager on z/OS

When you have customized or migrated your queue manager, you can test it by running the installation verification programs and some of the sample applications shipped with IBM MQ for z/OS.

About this task

After you have installed and customized IBM MQ for z/OS, you can use the supplied installation verification program, CSQ4IVP1, to confirm that IBM MQ for z/OS is operational.

The basic installation verification program CSQ4IVP1 tests non-shared queues and verifies the base IBM MQ without using the C, COBOL, or CICS samples.

After running the basic installation verification, you can test for shared queues by using CSQ4IVP1 with different queues, and also test that Db2 and the coupling facility are set up correctly. To confirm that distributed queuing is operational, you can use the supplied installation verification program, CSQ4IVPX,

CSQ4IVP1 is supplied as a load module, and provides a set of procedural sample applications as source modules that demonstrate typical uses of the Message Queue Interface (MQI). You can use these source modules to test different programming language environments. You can compile and link-edit whichever of the other samples are appropriate to your installation by using the supplied sample JCL supplied.

Procedure

- For information on how to test your queue manager on z/OS, see the following subtopics:
  - “Running the basic installation verification program” on page 967
  - “Testing for queue sharing groups” on page 971
  - “Testing for distributed queuing” on page 972
  - “Testing for C, C++, COBOL, PL/I, and CICS programs with IBM MQ for z/OS” on page 975

Related concepts

IBM MQ for z/OS concepts

Related tasks

Planning your IBM MQ environment on z/OS

“Configuring queue managers on z/OS” on page 896

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

Administering IBM MQ for z/OS

z/OS Running the basic installation verification program

After you have installed and customized IBM MQ, you can use the supplied installation verification program, CSQ4IVP1, to confirm that IBM MQ is operational.

The basic installation verification program is a batch assembler IVP that verifies the base IBM MQ without using the C, COBOL, or CICS samples.

The Batch Assembler IVP is link-edited by SMP/E and the load modules are shipped in library thlqual.SCSQLOAD.

After you have completed both the SMP/E APPLY step and the customization steps, run the Batch Assembler IVP.

See these sections for further details:

- Overview of the CSQ4IVP1 application
- Preparing to run CSQ4IVP1
- Running CSQ4IVP1

- Checking the results of CSQ4IVP1

Overview of the CSQ4IVP1 application

CSQ4IVP1 is a batch application that connects to your IBM MQ subsystem and performs these basic functions:

- Issues IBM MQ calls
- Communicates with the command server
- Verifies that triggering is active
- Generates and deletes a dynamic queue
- Verifies message expiry processing
- Verifies message commit processing

Preparing to run CSQ4IVP1

Before you run CSQ4IVP1:

1. Check that the IVP entries are in the CSQINP2 data set concatenation in the queue manager startup program. The IVP entries are supplied in member thlqual.SCSQPROC(CSQ4IVPQ). If not, add the definitions supplied in thlqual.SCSQPROC(CSQ4IVPQ) to your CSQINP2 concatenation. If the queue manager is currently running, you need to restart it so that these definitions can take effect.
2. The sample JCL, CSQ4IVPR, required to run the installation verification program is in library thlqual.SCSQPROC.

   Customize the CSQ4IVPR JCL with the high-level qualifier for the IBM MQ libraries, the national language you want to use, the four-character IBM MQ queue manager name, and the destination for the job output.
3. Update RACF to allow CSQ4IVP1 to access its resources if IBM MQ security is active.

   To run CSQ4IVP1 when IBM MQ security is enabled, you need a RACF user ID with authority to access the objects. For details of defining resources to RACF, see Setting up security on z/OS . The user ID that runs the IVP must have the following access authority:

Authority	Profile	Class
READ	ssid.DISPLAY.PROCESS	MQCMDS
UPDATE	ssid.SYSTEM.COMMAND.INPUT	MQQUEUE
UPDATE	ssid.SYSTEM.COMMAND.REPLY.MODEL	MQQUEUE
UPDATE	ssid.CSQ4IVP1.**	MQQUEUE
READ	ssid.BATCH	MQCONN

   These requirements assume that all IBM MQ security is active. The RACF commands to activate IBM MQ security are shown in Figure 98 on page 969. This example assumes that the queue manager name is CSQ1 and that the user ID of the person running sample CSQ4IVP1 is TS101.

```
RDEFINE MQCMDS CSQ1.DISPLAY.PROCESS
PERMIT CSQ1.DISPLAY.PROCESS CLASS(MQCMDS) ID(TS101) ACCESS(READ)

RDEFINE MQQUEUE CSQ1.SYSTEM.COMMAND.INPUT
PERMIT CSQ1.SYSTEM.COMMAND.INPUT CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.COMMAND.REPLY.MODEL
PERMIT CSQ1.SYSTEM.COMMAND.REPLY.MODEL CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.CSQ4IVP1.**
PERMIT CSQ1.CSQ4IVP1.** CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQCONN CSQ1.BATCH
PERMIT CSQ1.BATCH CLASS(MQCONN) ID(TS101) ACCESS(READ)
```

Figure 98. RACF commands for CSQ4IVP1

Running CSQ4IVP1

When you have completed these steps, start your queue manager. If the queue manager is already running and you have changed CSQINP2, you must stop the queue manager and restart it.

The IVP runs as a batch job. Customize the job card to meet the submission requirements of your installation.

Checking the results of CSQ4IVP1

The IVP is split into 10 stages; each stage must complete with a zero completion code before the next stage is run. The IVP generates a report, listing:

- The name of queue manager that is being connected to.
- A one-line message showing the completion code and the reason code returned from each stage.
- A one-line informational message where appropriate.

A sample report is provided in Figure 99 on page 971

z/OS For an explanation of the completion and reason codes, see the IBM MQ for z/OS messages, completion, and reason codes.

Some stages have more than one IBM MQ call and, in the event of failure, a message is issued indicating the specific IBM MQ call that returned the failure. Also, for some stages the IVP puts explanatory and diagnostic information into a comment field.

The IVP job requests exclusive control of certain queue manager objects and therefore should be single threaded through the system. However, there is no limit to the number of times the IVP can be run against your queue manager.

The functions performed by each stage are:

Stage 1
Connect to the queue manager by issuing the MQCONN API call.

Stage 2
Determine the name of the system-command input queue used by the command server to retrieve request messages. This queue receives display requests from Stage 5.

To do this, the sequence of calls is:

1. Issue an MQOPEN call, specifying the queue manager name, to open the queue manager object.
2. Issue an MQINQ call to find out the name of the system-command input queue.
3. Issue an MQINQ call to find out about various queue manager event switches.
4. Issue an MQCLOSE call to close the queue manager object.

On successful completion of this stage, the name of the system-command input queue is displayed in the comment field.

Stage 3
Open an initiation queue using an **MQOPEN** call.

This queue is opened at this stage in anticipation of a trigger message, which arrives as a result of the command server replying to the request from Stage 5. The queue must be opened for input to meet the triggering criteria.

Stage 4
Create a permanent dynamic queue using the CSQ4IVP1.MODEL queue as a model. The dynamic queue has the same attributes as the model from which it was created. This means that when the replies from the command server request in Stage 5 are written to this queue, a trigger message is written to the initiation queue opened in Stage 3.

Upon successful completion of this stage, the name of the permanent dynamic queue is indicated in the comment field.

Stage 5
Issue an MQPUT1 request to the command server command queue.

A message of type MQMT_REQUEST is written to the system-command input queue requesting a display of process CSQ4IVP1. The message descriptor for the message specifies the permanent dynamic queue created in Stage 4 as the reply-to queue for the command server's response.

Stage 6
Issue an **MQGET** request from the initiation queue. At this stage, a GET WAIT with an interval of 1 minute is issued against the initiation queue opened in Stage 3. The message returned is expected to be the trigger message generated by the command server's response messages being written to the reply-to queue.

Stage 7
Delete the permanent dynamic queue created in Stage 4. As the queue still has messages on it, the MQCO_PURGE_DELETE option is used.

Stage 8

1. Open a dynamic queue.
2. MQPUT a message with an expiry interval set.
3. Wait for the message to expire.
4. Attempt to MQGET the expired message.
5. MQCLOSE the queue.

Stage 9

1. Open a dynamic queue.
2. MQPUT a message.
3. Issue MQCMIT to commit the current unit of work.
4. MQGET the message.
5. Issue MQBACK to backout the message.
6. MQGET the same message and ensure that the backout count is set to 1.
7. Issue MQCLOSE to close the queue.

Stage 10
Disconnect from the queue manager using **MQDISC**.

After running the IVP, you can delete any objects that you no longer require.

If the IVP does not run successfully, try each step manually to find out which function is failing.

```
DATE : 2005.035                IBM MQ for z/OS - V6           PAGE : 0001
INSTALLATION VERIFICATION PROGRAM
PARAMETERS ACCEPTED. PROGRAM WILL CONNECT TO : CSQ1
,OBJECT QUALIFER : CSQ4IVP1
INSTALLATION VERIFICATION BEGINS :
STAGE 01 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR BRIDGE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS EXCP FOR CHANNEL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR SSL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR INHIBITED EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR LOCAL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR PERFORMENCE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR REMOTE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR START/STOP EVENTS
STAGE 02 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 SYSTEM.COMMAND.INPUT
STAGE 03 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 04 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 CSQ4IVP1.BAB9810EFEAC8980
STAGE 05 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 06 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 07 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 08 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 CSQ4IVP1.BAB9810F0070E645
STAGE 09 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 CSQ4IVP1.BAB9812BA8706803
STAGE 10 COMPLETE. COMPCODE : 0000 REASON CODE : 0000>>>>>>>>>>   END OF REPORT   <<<<<<<<<<
```

Figure 99. Sample report from CSQ4IVP1

z/OS Testing for queue sharing groups

The basic installation verification program CSQ4IVP1 tests non-shared queues.

CSQ4IVP1 can be used whether the queue manager is a member of a queue sharing group or not. After running the basic IVP, you can test for shared queues by using the CSQ4IVP1 installation verification program with different queues. Also this tests that Db2 and the coupling facility are set up correctly.

Preparing to run CSQ4IVP1 for a queue sharing group

Before you run CSQ4IVP1:

1. Add the coupling facility structure that the IVP uses to your CFRM policy data set, as described in "Set up the coupling facility" on page 954. The supplied samples use a structure called APPLICATION1, but you can change this if you want.
2. Check that the IVP entries are in the CSQINP2 data set concatenation in the queue manager startup program. The IVP entries are supplied in member thlqual.SCSQPROC(CSQ4IVPG). If they are not, add the definitions supplied in thlqual.SCSQPROC(CSQ4IVPG) to your CSQINP2 concatenation. If the queue manager is currently running, you need to restart it so that these definitions can take effect.
3. Change the name of the coupling facility structure used in thlqual.SCSQPROC(CSQ4IVPG) if necessary.
4. The sample JCL, CSQ4IVPS, required to run the installation verification program for a queue sharing group is in library thlqual.SCSQPROC.

   Customize the CSQ4IVPS JCL with the high-level qualifier for the IBM MQ libraries, the national language you want to use, the four-character IBM MQ queue manager name, and the destination for the job output.
5. Update RACF to allow CSQ4IVP1 to access its resources if IBM MQ security is active.

   To run CSQ4IVP1 when IBM MQ security is enabled, you need a RACF user ID with authority to access the objects. For details of defining resources to RACF, see Setting up security on z/OS . The user ID that runs the IVP must have the following access authority in addition to that required to run the basic IVP:

Authority	Profile	Class
UPDATE	ssid.CSQ4IVPG.**	MQQUEUE

These requirements assume that all IBM MQ security is active. The RACF commands to activate IBM MQ security are shown in Figure 100 on page 972. This example assumes that the queue manager name is CSQ1 and that the user ID of the person running sample CSQ4IVP1 is TS101.

```
RDEFINE MQQUEUE CSQ1.CSQ4IVPG.**
PERMIT CSQ1.CSQ4IVPG.** CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)
```

Figure 100. RACF commands for CSQ4IVP1 for a queue sharing group

Running CSQ4IVP1 for a queue sharing group

When you have completed these steps, start your queue manager. If the queue manager is already running and you have changed CSQINP2, you must stop the queue manager and restart it.

The IVP runs as a batch job. Customize the job card to meet the submission requirements of your installation.

Checking the results of CSQ4IVP1 for a queue sharing group

The IVP for queue sharing groups works in the same way as the basic IVP, except that the queues that are created are called CSQIVPG. *xx*. Follow the instructions given in "Checking the results of CSQ4IVP1" on page 969 to check the results of the IVP for queue sharing groups.

z/OS Testing for distributed queuing

You can use the supplied installation verification program, CSQ4IVPX, to confirm that distributed queuing is operational.

Overview of CSQ4IVPX job

CSQ4IVPX is a batch job that starts the channel initiator and issues the IBM MQ DISPLAY CHINIT command. This verifies that all major aspects of distributed queuing are operational, while avoiding the need to set up channel and network definitions.

Preparing to run CSQ4IVPX

Before you run CSQ4IVPX:

1. The sample JCL, CSQ4IVPX, required to run the installation verification program is in library thlqual.SCSQPROC.

   Customize the CSQ4IVPX JCL with the high-level qualifier for the IBM MQ libraries, the national language you want to use, the four-character queue manager name, and the destination for the job output.
2. Update RACF to allow CSQ4IVPX to access its resources if IBM MQ security is active. To run CSQ4IVPX when IBM MQ security is enabled, you need a RACF user ID with authority to access the objects. For details of defining resources to RACF, see Setting up security on z/OS . The user ID that runs the IVP must have the following access authority:

Authority	Profile	Class
CONTROL	ssid.START.CHINIT and ssid.STOP.CHINIT	MQCMDS
UPDATE	ssid.SYSTEM.COMMAND.INPUT	MQQUEUE
UPDATE	ssid.SYSTEM.CSQUTIL.*	MQQUEUE
READ	ssid.BATCH	MQCONN

Authority	Profile	Class
READ	ssid.DISPLAY.CHINIT	MQCMDS

These requirements assume that the connection security profile ssid.CHIN has been defined (as shown in Connection security profiles for the channel initiator), and that all IBM MQ security is active. The RACF commands to do this are shown in Figure 101 on page 974. This example assumes that:

- The queue manager name is CSQ1
- The user ID of the person running sample CSQ4IVPX is TS101
- The channel initiator address space is running under the user ID CSQ1MSTR

3. Update RACF to allow the channel initiator address space the following access authority:

Authority	Profile	Class
READ	ssid.CHIN	MQCONN
UPDATE	ssid.SYSTEM.COMMAND.INPUT	MQQUEUE
UPDATE	ssid.SYSTEM.CHANNEL.INITQ	MQQUEUE
UPDATE	ssid.SYSTEM.CHANNEL.SYNCQ	MQQUEUE
ALTER	ssid.SYSTEM.CLUSTER.COMMAND.QUEUE	MQQUEUE
UPDATE	ssid.SYSTEM.CLUSTER.TRANSMIT.QUEUE	MQQUEUE
ALTER	ssid.SYSTEM.CLUSTER.REPOSITORY.QUEUE	MQQUEUE
CONTROL	ssid.CONTEXT.**	MQADMIN

The RACF commands to do this are also shown in Figure 101 on page 974.

```
RDEFINE MQCMDS CSQ1.DISPLAY.DQM
PERMIT CSQ1.DISPLAY.DQM CLASS(MQCMDS) ID(TS101) ACCESS(READ)

RDEFINE MQCMDS CSQ1.START.CHINIT
PERMIT CSQ1.START.CHINIT CLASS(MQCMDS) ID(TS101) ACCESS(CONTROL)

RDEFINE MQCMDS CSQ1.STOP.CHINIT
PERMIT CSQ1.STOP.CHINIT CLASS(MQCMDS) ID(TS101) ACCESS(CONTROL)

RDEFINE MQQUEUE CSQ1.SYSTEM.COMMAND.INPUT
PERMIT CSQ1.SYSTEM.COMMAND.INPUT CLASS(MQQUEUE) ID(TS101,CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.CSQUTIL.*
PERMIT CSQ1.SYSTEM.CSQUTIL.* CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQCONN CSQ1.BATCH
PERMIT CSQ1.BATCH CLASS(MQCONN) ID(TS101) ACCESS(READ)

RDEFINE MQCONN CSQ1.CHIN
PERMIT CSQ1.CHIN CLASS(MQCONN) ID(CSQ1MSTR) ACCESS(READ)

RDEFINE MQQUEUE CSQ1.SYSTEM.CHANNEL.SYNCQ
PERMIT CSQ1.SYSTEM.CHANNEL.SYNCQ CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.CLUSTER.COMMAND.QUEUE
PERMIT CSQ1.SYSTEM.CLUSTER.COMMAND.QUEUE CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(ALTER)

RDEFINE MQQUEUE CSQ1.SYSTEM.CLUSTER.TRANSMIT.QUEUE
PERMIT CSQ1.SYSTEM.CLUSTER.TRANSMIT.QUEUE CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.CLUSTER.REPOSITORY.QUEUE
PERMIT CSQ1.SYSTEM.CLUSTER.REPOSITORY.QUEUE CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(ALTER)

RDEFINE MQQUEUE CSQ1.SYSTEM.CHANNEL.INITQ
PERMIT CSQ1.SYSTEM.CHANNEL.INITQ CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQADMIN CSQ1.CONTEXT.**
PERMIT CSQ1.CONTEXT.** CLASS(MQADMIN) ID(CSQ1MSTR) ACCESS(CONTROL)
```

Figure 101. RACF commands for CSQ4IVPX

Running CSQ4IVPX

When you have completed these steps, start your queue manager.

The IVP runs as a batch job. Customize the job card to meet the submission requirements of your installation.

Checking the results of CSQ4IVPX

CSQ4IVPX runs the CSQUTIL IBM MQ utility to issue three MQSC commands. The SYSPRINT output data set should look like Figure 102 on page 975, although details might differ depending on your queue manager attributes.

- You should see the commands **(1)** each followed by several messages.
- The last message from each command should be "CSQ9022I ... NORMAL COMPLETION" **(2)**.
- The job as a whole should complete with return code zero **(3)**.

```
CSQU000I CSQUTIL IBM MQ for z/OS - V6
CSQU001I CSQUTIL Queue Manager Utility - 2005-05-09 09:06:48
COMMAND
CSQU127I CSQUTIL Executing COMMAND using input from CSQUCMD data set
CSQU120I CSQUTIL Connecting to queue manager CSQ1
CSQU121I CSQUTIL Connected to queue manager CSQ1
CSQU055I CSQUTIL Target queue manager is CSQ1
START CHINIT
(1)
CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000004
CSQM138I +CSQ1 CSQMSCHI CHANNEL INITIATOR STARTING
CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000000
CSQ9022I +CSQ1 CSQXCRPS ' START CHINIT' NORMAL COMPLETION
(2)
DISPLAY CHINIT
(1)
CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000004
CSQM137I +CSQ1 CSQMDDQM DISPLAY CHINIT COMMAND ACCEPTED
CSQN205I  COUNT=   12, RETURN=00000000, REASON=00000000
CSQX830I +CSQ1 CSQXRDQM Channel initiator active
CSQX002I +CSQ1 CSQXRDQM Queue sharing group is QSG1
CSQX831I +CSQ1 CSQXRDQM 8 adapter subtasks started, 8 requested
CSQX832I +CSQ1 CSQXRDQM 5 dispatchers started, 5 requested
CSQX833I +CSQ1 CSQXRDQM 0 SSL server subtasks started, 0 requested
CSQX840I +CSQ1 CSQXRDQM 0 channel connections current, maximum 200
CSQX841I +CSQ1 CSQXRDQM 0 channel connections active, maximum 200,
including 0 paused
CSQX842I +CSQ1 CSQXRDQM 0 channel connections starting,
0 stopped, 0 retrying
CSQX836I +CSQ1 Maximum channels - TCP/IP 200, LU 6.2 200
CSQX845I +CSQ1 CSQXRDQM TCP/IP system name is TCPIP
CSQX848I +CSQ1 CSQXRDQM TCP/IP listener INDISP=QMGR not started
CSQX848I +CSQ1 CSQXRDQM TCP/IP listener INDISP=GROUP not started
CSQX849I +CSQ1 CSQXRDQM LU 6.2 listener INDISP=QMGR not started
CSQX849I +CSQ1 CSQXRDQM LU 6.2 listener INDISP=GROUP not started
CSQ9022I +CSQ1 CSQXCRPS ' DISPLAY CHINIT' NORMAL COMPLETION
(2)
STOP CHINIT
(1)
CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000004
CSQM137I +CSQ1 CSQMTCHI STOP CHINIT COMMAND ACCEPTED
CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000000
CSQ9022I +CSQ1 CSQXCRPS ' STOP CHINIT' NORMAL COMPLETION
(2)
CSQU057I CSQUCMDS 3 commands read
CSQU058I CSQUCMDS 3 commands issued and responses received, 0 failed
CSQU143I CSQUTIL 1 COMMAND statements attempted
CSQU144I CSQUTIL 1 COMMAND statements executed successfully
CSQU148I CSQUTIL Utility completed, return code=0
(3)
```

Figure 102. Example output from CSQ4IVPX

z/OS Testing for C, C++, COBOL, PL/I, and CICS programs with IBM MQ for z/OS

You can test for C, C++, COBOL, PL/I, or CICS, using the sample applications supplied with IBM MQ.

The IVP (CSQ4IVP1) is supplied as a load module, and provides the samples as source modules. You can use these source modules to test different programming language environments.

For more information about sample applications, see Sample programs for IBM MQ for z/OS .

z/OS Setting up communications with other queue managers on z/OS

This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

About this task

To define your distributed-queuing requirements, you need to define the following items:

- The channel initiator procedures and data sets
- The channel definitions
- The queues and other objects
- Access security

If you are using queue sharing groups, see Distributed queuing and queue sharing groups.

For additional points to consider when you are preparing to set up distributed queuing with IBM MQ for z/OS, see “Considerations for using distributed queuing on z/OS” on page 976.

Procedure

To enable distributed queuing, complete the following steps:

- Customize the distributed queuing facility and define the IBM MQ objects required as described in Defining system objects and “Preparing to customize queue managers on z/OS” on page 897.
- Define access security as described in Security considerations for the channel initiator on z/OS .
- Set up your communications as described in “Setting up communication for z/OS” on page 995.

Related concepts

“Setting up IBM MQ for z/OS” on page 901
Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

Related tasks

“Configuración de la gestión de colas distribuidas” on page 210
En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

z/OS Considerations for using distributed queuing on z/OS

Points to consider when you are preparing to use distributed queuing on z/OS.

If you are using queue sharing groups, see Distributed queuing and queue sharing groups.

Operator messages

Because the channel initiator uses a number of asynchronously operating dispatchers, operator messages might occur on the log out of chronological sequence.

Channel operation commands

Channel operation commands generally involve two stages. When the command syntax has been checked and the existence of the channel verified, a request is sent to the channel initiator. Message CSQM134I or CSQM137I is sent to the command issuer to indicate the completion of the first stage. When the channel initiator has processed the command, further messages indicating its success or otherwise are sent to the command issuer along with message CSQ9022I or CSQ9023E. Any error messages generated could also be sent to the z/OS console.

All cluster commands except **DISPLAY CLUSQMGR**, however, work asynchronously. Commands that change object attributes update the object and send a request to the channel initiator. Commands for working with clusters are checked for syntax and a request is sent to the channel initiator. In both cases, message CSQM130I is sent to the command issuer indicating that a request has been sent. This message is followed by message CSQ9022I to indicate that the command has completed successfully, in that a request has been sent. It does not indicate that the cluster request has completed successfully. The requests sent to the channel initiator are processed asynchronously, along with cluster requests received

from other members of the cluster. In some cases, these requests must be sent to the whole cluster to determine if they are successful or not. Any errors are reported to the z/OS on the system where the channel initiator is running. They are not sent to the command issuer.

Undelivered-message queue

A Dead Letter handler is provided with IBM MQ for z/OS. For more information, see The dead-letter queue handler utility (CSQUDLQH).

Queues in use

MCAs for receiver channels can keep the destination queues open even when messages are not being transmitted. This behavior results in the queues appearing to be 'in use'.

Security changes

If you change security access for a user ID, the change might not take effect immediately. For more information, see Security considerations for the channel initiator on z/OS , Profiles for queue security, and “Implement your ESM security controls” on page 915.

Communications stopped - TCP

If TCP is stopped for some reason and then restarted, the IBM MQ for z/OS TCP listener waiting on a TCP port is stopped.

Automatic channel-reconnect allows the channel initiator to detect that TCP/IP is unavailable and to automatically restart the TCP/IP listener when TCP/IP returns. This automatic restart alleviates the need for operations staff to notice the problem with TCP/IP and manually restart the listener. While the listener is out of action, the channel initiator can also be used to try the listener again at the interval specified by LSTRTMR. These attempts can continue until TCP/IP returns and the listener successfully restarts automatically. For more information about LSTRTMR, see ALTER QMGR and Distributed queuing messages (CSQX...).

Communications stopped - LU6.2

If APPC is stopped, the listener is also stopped. Again, in this case, the listener automatically tries again at the LSTRTMR interval so that, if APPC restarts, the listener can restart too.

If the Db2 fails, shared channels that are already running continue to run, but any new channel start requests fail. When the Db2 is restored new requests are able to complete.

z/OS Automatic Restart Management (ARM)

Automatic restart management (ARM) is a z/OS recovery function that can improve the availability of specific batch jobs or started tasks (for example, subsystems). It can therefore result in a faster resumption of productive work.

To use ARM, you must set up your queue managers and channel initiators in a particular way to make them restart automatically. For more information, see Using the z/OS Automatic Restart Manager (ARM).

Related concepts

“Setting up IBM MQ for z/OS” on page 901
Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

Related tasks

“Configuración de la gestión de colas distribuidas” on page 210
En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

z/OS Defining IBM MQ objects on z/OS

On z/OS, use one of the IBM MQ command input methods to define IBM MQ objects.

For more information about defining objects, see "Monitoring and controlling channels on z/OS" on page 979.

Transmission queues and triggering channels

Define the following:

- A local queue with the usage of XMITQ for each sending message channel.
- Remote queue definitions.

  A remote queue object has three distinct uses, depending upon the way the name and content are specified:

  – Remote queue definition
  – Queue manager alias definition
  – Reply-to queue alias definition

  These three ways are shown in Three ways of using the remote queue definition object.

Use the TRIGDATA field on the transmission queue to trigger the specified channel. For example:

```
DEFINE QLOCAL(MYXMITQ) USAGE(XMITQ) TRIGGER +
INITQ(SYSTEM.CHANNEL.INITQ) TRIGDATA(MYCHANNEL)
DEFINE CHL(MYCHANNEL) CHLTYPE(SDR) TRPTYPE(TCP) +
XMITQ(MYXMITQ) CONNAME('9.20.9.30(1555)')
```

The supplied sample CSQ4INYD gives additional examples of the necessary definitions.

z/OS Loss of connectivity to the CF structure where the synchronization queue for shared channels is defined, or similar problems, might temporarily prevent a channel from starting. After problem resolution, if you are using a trigger type of FIRST and the channel fails to start when it is triggered, you must start the channel manually. If you want to automatically start triggered channels after problem resolution, consider setting the queue manager TRIGINT attribute to a value other than the default. Setting the TRIGINT attribute to a value other than the default causes the channel initiator to retry starting the channel periodically while there are messages on the transmission queue.

Synchronization queue

DQM requires a queue for use with sequence numbers and logical units of work identifiers (LUWID). You must ensure that a queue is available with the name SYSTEM.CHANNEL.SYNCQ (see Planificación en z/OS). This queue must be available otherwise the channel initiator cannot start.

Make sure that you define this queue using INDXTYPE(MSGID). This attribute improves the speed at which they can be accessed.

Channel command queues

You need to ensure that a channel command queue exists for your system with the name SYSTEM.CHANNEL.INITQ.

If the channel initiator detects a problem with the SYSTEM.CHANNEL.INITQ, it is unable to continue normally until the problem is corrected. The problem could be one of the following:

- The queue is full
- The queue is not enabled for put
- The page set that the queue is on is full
- The channel initiator does not have the correct security authorization to the queue

If the definition of the queue is changed to GET(DISABLED) while the channel initiator is running, the initiator is unable to get messages from the queue, and terminates.

Starting the channel initiator

Triggering is implemented using the channel initiator. On IBM MQ for z/OS, the initiator is started with the MQSC command `START CHINIT`.

Stopping the channel initiator

The channel initiator is stopped automatically when you stop the queue manager. If you need to stop the channel initiator but not the queue manager, you can use the MQSC command `STOP CHINIT`.

z/OS Monitoring and controlling channels on z/OS

Use the DQM commands and panels to create, monitor, and control the channels to remote queue managers.

Each z/OS queue manager has a DQM program (the *channel initiator*) for controlling interconnections to remote queue managers using native z/OS facilities.

The implementation of these panels and commands on z/OS is integrated into the operations and control panels and the MQSC commands. No differentiation is made in the organization of these two sets of panels and commands.

You can also enter commands using Programmable Command Format (PCF) commands. See Automating administration tasks for information about using these commands.

The information in this section applies in all cases where the channel initiator is used for distributed queuing. It applies whether you are using queue sharing groups, or intra-group queuing.

The DQM channel control function

For an overview of the distributed queue management model, see “Envío y recepción de mensajes” on page 232.

The channel control function consists of panels, commands and programs, two synchronization queues, channel command queues, and the channel definitions. This topic is a brief description of the components of the channel control function.

- The channel definitions are held as objects in page set zero or in Db2, like other IBM MQ objects in z/OS.
- You use the operations and control panels, MQSC commands, or PCF commands to:
  - Create, copy, display, alter, and delete channel definitions
  - Start and stop channel initiators and listeners
  - Start, stop, and ping channels, reset channel sequence numbers, and resolve in-doubt messages when links cannot be re-established
  - Display status information about channels
  - Display information about DQM

  In particular, you can use the CSQINPX initialization input data set to issue your MQSC commands. This set can be processed every time you start the channel initiator. For more information, see Initialization commands.
- There are two queues (SYSTEM.CHANNEL.SYNCQ and SYSTEM.QSG.CHANNEL.SYNCQ) used for channel re-synchronization purposes. Define these queues with INDXTYPE(MSGID) for performance reasons.
- The channel command queue (SYSTEM.CHANNEL.INITQ) is used to hold commands for channel initiators, channels, and listeners.
- The channel control function program runs in its own address space, separate from the queue manager, and comprises the channel initiator, listeners, MCAs, trigger monitor, and command handler.

- For queue sharing groups and shared channels, see Shared queues and queue sharing groups.
- For intra-group queuing, see Intra-group queuing

Managing your channels on z/OS

Use the links in the following table for information about how to manage your channels, channel initiators, and listeners:

Table 65. Channel tasks

Task to be performed	**MQSC command**
Define a channel	DEFINE CHANNEL
Alter a channel definition	ALTER CHANNEL
Display a channel definition	DISPLAY CHANNEL
Delete a channel definition	DELETE CHANNEL
Start a channel initiator	START CHINIT
Stop a channel initiator	STOP CHINIT
Display channel initiator information	DISPLAY CHINIT
Start a channel listener	START LISTENER
Stop a channel listener	STOP LISTENER
Start a channel	START CHANNEL
Test a channel	PING CHANNEL
Reset message sequence numbers for a channel	RESET CHANNEL
Resolve in-doubt messages on a channel	RESOLVE CHANNEL
Stop a channel	STOP CHANNEL
Display channel status	DISPLAY CHSTATUS
Display cluster channels	DISPLAY CLUSQMGR

Related concepts

"Using the panels and the commands" on page 981
You can use the MQSC commands, the PCF commands, or the operations and control panels to manage DQM.

"Setting up IBM MQ for z/OS" on page 901
Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

"Setting up communication for z/OS" on page 995
When a distributed-queuing management channel is started, it tries to use the connection specified in the channel definition. To succeed, it is necessary for the connection to be defined and available. This section explains how to define a connection.

"Preparing IBM MQ for z/OS for DQM with queue sharing groups" on page 1000
Use the instructions in this section to configure distributed queuing with queue sharing groups on IBM MQ for z/OS.

"Setting up communication for IBM MQ for z/OS using queue sharing groups" on page 1004
When a distributed-queuing management channel is started, it attempts to use the connection specified in the channel definition. For this attempt to succeed, it is necessary for the connection to be defined and available.

Related tasks

"Configuración de la gestión de colas distribuidas" on page 210
En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

"Setting up communications with other queue managers on z/OS" on page 975
This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

z/OS *Using the panels and the commands*

You can use the MQSC commands, the PCF commands, or the operations and control panels to manage DQM.

For information about MQSC commands, see Administering IBM MQ using MQSC commands. For information about PCF commands, see Automating administration using Programmable Command Formats commands.

Using the initial panel

For an introduction to invoking the operations and control panels, using the function keys, and getting help, see Administración IBM MQ for z/OS.

Note: To use the operations and control panels, you must have the correct security authorization; see Administering IBM MQ for z/OS and sub topics for more information. Figure 103 on page 981 shows the panel that is displayed when you start a panel session. The text after the panel explains the actions you perform in this panel.

```
IBM MQ for z/OS - Main Menu

Complete fields. Then press Enter.

Action . . . . . . . . . . 1   0. List with filter  4. Manage
1. List or Display  5. Perform
2. Define like    6. Start
3. Alter       7. Stop
8. Command
Object type . . . . . . . . CHANNEL    +
Name . . . . . . . . . . . *
Disposition . . . . . . . . A Q=Qmgr, C=Copy, P=Private, G=Group,
S=Shared, A=All

Connect name . . . . . . . MQ25 - local queue manager or group
Target queue manager . . . MQ25
- connected or remote queue manager for command input
Action queue manager . . . MQ25 - command scope in group
Response wait time . . . . 10  5 - 999 seconds

(C) Copyright IBM Corporation 1993, 2024. All rights reserved.

Command ===> ____________________________________________________________
F1=Help   F2=Split   F3=Exit   F4=Prompt  F9=SwapNext F10=Messages
F12=Cancel
```

Figure 103. The operations and controls initial panel

From this panel, you can:

- Select the action you want to perform by typing in the appropriate number in the **Action** field.
- Specify the object type that you want to work with. Press F4 for a list of object types if you are not sure what they are.
- Display a list of objects of the type specified. Type in an asterisk (*) in the **Name** field and press enter to display a list of objects (of the type specified) that have already been defined on this subsystem. You

can then select one or more objects to work with in sequence. Figure 104 on page 982 shows a list of channels produced in this way.

- Specify the disposition in the queue sharing group of the objects you want to work with in the **Disposition** field. The disposition determines where the object is kept and how the object behaves.
- Choose the local queue manager, or queue sharing group to which you want to connect in the **Connect name** field. If you want the commands to be issued on a remote queue manager, choose either the **Target queue manager** field or the **Action queue manager** field, depending upon whether the remote queue manager is not or is a member of a queue sharing group. If the remote queue manager is not a member of a queue sharing group, choose the **Target queue manager** field. If the remote queue manager is a member of a queue sharing group, choose the **Action queue manager** field.
- Choose the wait time for responses to be received in the **Response wait time** field.

```
List Channels - MQ25          Row 1 of 8

Type action codes, then press Enter. Press F11 to display connection status.
1=Display  2=Define like  3=Alter  4=Manage  5=Perform
6=Start   7=Stop

Name         Type     Disposition  Status
<> *           CHANNEL    ALL   MQ25
_  SYSTEM.DEF.CLNTCONN  CLNTCONN   QMGR  MQ25
_  SYSTEM.DEF.CLUSRCVR  CLUSRCVR   QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.CLUSSDR  CLUSSDR    QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.RECEIVER  RECEIVER   QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.REQUESTER REQUESTER   QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.SENDER   SENDER    QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.SERVER   SERVER    QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.SVRCONN  SVRCONN    QMGR  MQ25 INACTIVE
******** End of list ********



Command ===> ____________________________________________________________
F1=Help   F2=Split   F3=Exit   F4=Filter  F5=Refresh  F7=Bkwd
F8=Fwd    F9=SwapNext F10=Messages F11=Status  F12=Cancel
```

Figure 104. Listing channels

z/OS *Defining a channel on z/OS*

On z/OS, you can define a channel by using MQSC commands or using the operations and control panels.

Procedure

- To define a channel using the MQSC commands, use the **DEFINE CHANNEL** command.
- To use the operations and control panels, starting from the initial panel, complete the following fields and press Enter:

Table 66. Operations and control panels: initial panel fields

Field	Value to enter in field
Action	2 (Define like)
Object type	Channel type (for example SENDER) or CHANNEL
Name	
Disposition	The location of the new object.

You are presented with some panels to complete with information about the name and attributes that you want for the channel you are defining. They are initialized with the default attribute values. Change any that you want to before pressing Enter.

Note: If you entered CHANNEL in the **object type** field, you are presented with the **Select a Valid Channel Type** panel first.

If you want to define a channel with the same attributes as an existing channel, put the name of the channel you want to copy in the **Name** field on the initial panel. The panels are initialized with the attributes of the existing object.

For information about the channel attributes, see Channel attributes.

Note:

1. Name all the channels in your network uniquely. As shown in Network diagram showing all channels, including the source and target queue manager names in the channel name is a good way to do this naming.

What to do next

After you have defined your channel you must secure your channel. For more information, see "Securing a channel" on page 984.

z/OS *Altering a channel definition*

You can alter a channel definition using MQSC commands or using the operations and control panels.

To alter a channel definition using the MQSC commands, use ALTER CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	3 (Alter)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.ALTER
Disposition	The location of the stored object.

You are presented with some panels containing information about the current attributes of the channel. Change any of the unprotected fields that you want by over typing the new value, and then press enter to change the channel definition.

For information about the channel attributes, see Channel attributes.

z/OS *Displaying a channel definition*

You can display a channel definition using MQSC commands or using the operations and control panels.

To display a channel definition using the MQSC commands, use DISPLAY CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	1 (List or Display)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.DISPLAY
Disposition	The location of the object.

You are presented with some panels displaying information about the current attributes of the channel.

For information about the channel attributes, see Channel attributes.

z/OS *Deleting a channel definition*

You can delete a channel definition using MQSC commands or using the operations and control panels.

To delete a channel definition using the MQSC commands, use DELETE CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	4 (Manage)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.DELETE
Disposition	The location of the object.

You are presented with another panel. Select function type 1 on this panel.

Press enter to delete the channel definition; you are asked to confirm that you want to delete the channel definition by pressing enter again.

Note: The channel initiator has to be running before a channel definition can be deleted (except for client-connection channels).

z/OS *Displaying information about the channel initiator*

You can display information about the channel initiator using MQSC commands or using the operations and control panels.

To display information about the channel initiator using the MQSC commands, use DISPLAY CHINIT.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	1 (Display)
Object type	SYSTEM
Name	Blank

You are presented with another panel. Select function type 1 on this panel.

Note:

1. Displaying distributed queuing information might take some time if you have lots of channels.
2. The channel initiator has to be running before you can display information about distributed queuing.

z/OS *Securing a channel*

You can secure a channel using MQSC commands or using the operations and control panels.

To secure a channel using the MQSC commands, use SET CHLAUTH.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	8

You are presented with an editor within which you can provide an MQSC command, in this case a CHLAUTH command, see Figure 105 on page 985. When you have finished typing in the command, the

plus signs (+) are needed. Type PF3 to exit from the editor and submit the command to the command server.

```
****************************** Top of Data ******************************
000001 SET CHLAUTH(SYSTEM.DEF.SVRCONN) +
000002 TYPE(SSLPEERMAP) +
000003 SSLPEER('CN="John Smith"') +
000004 MCAUSER('PUBLIC')
****** **************************** Bottom of Data ***************************

Command ===>                          Scroll ===> PAGE
F1=Help   F3=Exit   F4=LineEdit F12=Cancel
```

Figure 105. Command Entry

The output of the command is then presented to you, see Figure 106 on page 985

```
****** **************************** Top of Data ******************************
000001 CSQU000I CSQUTIL IBM MQ for z/OS V7.1.0
000002 CSQU001I CSQUTIL Queue Manager Utility - 2011-04-20 14:42:58
000003 COMMAND TGTQMGR(MQ23) RESPTIME(30)
000004 CSQU127I Executing COMMAND using input from CSQUCMD data set
000005 CSQU120I Connecting to MQ23
000006 CSQU121I Connected to queue manager MQ23
000007 CSQU055I Target queue manager is MQ23
000008 SET CHLAUTH(SYSTEM.DEF.SVRCONN) +
000009 TYPE(SSLPEERMAP) +
000010 SSLPEER('CN="John Smith"') +
000011 MCAUSER('PUBLIC')
000012 CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000000
000013 CSQ9022I !MQ23 CSQMSCA ' SET CHLAUTH' NORMAL COMPLETION
000014 CSQU057I 1 commands read
000015 CSQU058I 1 commands issued and responses received, 0 failed
000016 CSQU143I 1 COMMAND statements attempted
000017 CSQU144I 1 COMMAND statements executed successfully
000018 CSQU148I CSQUTIL Utility completed, return code=0
Command ===>                          Scroll ===> PAGE
F1=Help   F3=Exit   F5=Rfind   F6=Rchange  F9=SwapNext F12=Cancel
```

Figure 106. Command Output

z/OS *Starting a channel initiator*

You can start a channel initiator using MQSC commands or using the operations and control panels.

To start a channel initiator using the MQSC commands, use START CHINIT.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	6 (Start)
Object type	SYSTEM
Name	Blank

The Start a System Function panel is displayed. The text following the following panel explains what action to take:

```
Start a System Function

Select function type, complete fields, then press Enter to start system
function.

Function type . . . . . _   1. Channel initiator
2. Channel listener
Action queue manager . . . : MQ25

Channel initiator
JCL substitution . . . . . ________________________________________________
________________________________________

Channel listener
Inbound disposition . . . Q G=Group, Q=Qmgr
Transport type . . . . . _ L=LU6.2, T=TCP/IP
LU name (LU6.2) . . . . . ________________
Port number (TCP/IP) . . . 1414
IP address (TCP/IP) . . . ________________________________________________



Command ===> ____________________________________________________________
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

Figure 107. Starting a system function

Select function type 1 (channel initiator), and press enter.

z/OS *Stopping a channel initiator*

You can stop a channel initiator using MQSC commands or using the operations and control panels.

To stop a channel initiator using the MQSC commands, use STOP CHINIT.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	7 (Stop)
Object type	SYSTEM
Name	Blank

The Stop a System Function panel is displayed. The text following the panel explains how you to use this panel:

```
Stop a System Function

Select function type, complete fields, then press Enter to stop system
function.

Function type . . . . . _   1. Channel initiator
2. Channel listener
Action queue manager . . . : MQ25

Channel initiator
Restart shared channels  Y Y=Yes, N=No

Channel listener
Inbound disposition . . . Q G=Group, Q=Qmgr
Transport type . . . . . _ L=LU6.2, T=TCP/IP

Port number (TCP/IP) . . . _____
IP address (TCP/IP) . . . ________________________________________________



Command ===> ____________________________________________________________
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

Figure 108. Stopping a function control

Select function type 1 (channel initiator) and press enter.

The channel initiator waits for all running channels to stop in quiesce mode before it stops.

Note: If some of the channels are receiver or requester channels that are running but not active, a stop request issued to either the receiver or sender channel initiator causes it to stop immediately.

However, if messages are flowing, the channel initiator waits for the current batch of messages to complete before it stops.

z/OS *Starting a channel listener*

You can start a channel listener using MQSC commands or using the operations and control panels.

To start a channel listener using the MQSC commands, use START LISTENER.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	6 (Start)
Object type	SYSTEM
Name	Blank

The Start a System Function panel is displayed (see Figure 107 on page 986).

Select function type 2 (channel listener). Select Inbound disposition. Select Transport type. If the Transport type is L, select LU name. If the Transport type is T, select Port number and (optionally) IP address. Press enter.

Note: For the TCP/IP listener, you can start multiple combinations of Port and IP address.

z/OS *Stopping a channel listener*

You can stop a channel listener using MQSC commands or using the operations and control panels.

To stop a channel listener using the MQSC commands, use STOP LISTENER.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	7 (Stop)
Object type	SYSTEM
Name	Blank

The Stop a System Function panel is displayed (see Figure 108 on page 987).

Select function type 2 (channel listener). Select Inbound disposition. Select Transport type. If the transport type is 'T', select Port number and (optionally) IP address. Press enter.

Note: For a TCP/IP listener, you can stop specific combinations of Port and IP address, or you can stop all combinations.

z/OS *Starting a channel*

You can start a channel using MQSC commands or using the operations and control panels.

To start a channel using the MQSC commands, use START CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	6 (Start)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.USE
Disposition	The disposition of the object.

The Start a Channel panel is displayed. The text following the panel explains how to use the panel:

```
Start a Channel

Select disposition, then press Enter to start channel.


Channel name . . . . . : CHANNEL.TO.USE
Channel type . . . . . : SENDER
Description . . . . . : Description of CHANNEL.TO.USE


Disposition . . . . . P   P=Private on MQ25
S=Shared on MQ25
A=Shared on any queue manager









Command ===> ____________________________________________________________
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

Figure 109. Starting a channel

Select the disposition of the channel instance and on which queue manager it is to be started.

Press enter to start the channel.

z/OS *Starting a shared channel*

To start a shared channel, and keep it on a nominated channel initiator, use disposition = S (on the START CHANNEL command, specify CHLDISP(FIXSHARED)).

There can be only one instance of the shared channel running at a time. Attempts to start a second instance of the channel fail.

When you start a channel in this way, the following rules apply to that channel:

- You can stop the channel from any queue manager in the queue sharing group. You can stop it even if the channel initiator on which it was started is not running at the time you issue the stop-channel request. When the channel has stopped, you can restart it by specifying disposition = S (CHLDISP(FIXSHARED)) on the same, or another, channel initiator. You can also start it by specifying disposition = A (CHLDISP(SHARED)).
- If the channel is in the starting or retry state, you can restart it by specifying disposition = S (CHLDISP(FIXSHARED)) on the same or a different channel initiator. You can also start it by specifying disposition = A (CHLDISP(SHARED)).
- The channel is eligible to be trigger started when it goes into the inactive state. Shared channels that are trigger started always have a shared disposition (CHLDISP(SHARED)).
- The channel is eligible to be started with CHLDISP(FIXSHARED), on any channel initiator, when it goes into the inactive state. You can also start it by specifying disposition = A (CHLDISP(SHARED)).
- The channel is not recovered by any other active channel initiator in the queue sharing group when the channel initiator on which it was started is stopped with SHARED(RESTART), or when the channel initiator terminates abnormally. The channel is recovered only when the channel initiator on which it was started is next restarted. This stops failed channel-recovery attempts being passed to other channel initiators in the queue sharing group, which would add to their workload.

z/OS *Testing a channel*

You can test a channel using MQSC commands or using the operations and control panels.

To test a channel using the MQSC commands, use PING CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	5 (Perform)
Object type	SENDER, SERVER, or CHANNEL
Name	CHANNEL.TO.USE
Disposition	The disposition of the channel object.

The Perform a Channel Function panel is displayed. The text following the panel explains how to use the panel:

```
Perform a Channel Function

Select function type, complete fields, then press Enter.


Function type . . . . . _   1. Reset  3. Resolve with commit
2. Ping  4. Resolve with backout

Channel name . . . . . : CHANNEL.TO.USE
Channel type . . . . . : SENDER
Description . . . . . : Description of CHANNEL.TO.USE


Disposition . . . . . P   P=Private on MQ25
S=Shared on MQ25
A=Shared on any queue manager

Sequence number for reset . . 1     1 - 999999999
Data length for ping . . . . 16   16 - 32768



Command ===> ____________________________________________________________
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

Figure 110. Testing a channel

Select function type 2 (ping).

Select the disposition of the channel for which the test is to be done and on which queue manager it is to be tested.

The data length is initially set to 16. Change it if you want and press enter.

z/OS *Resetting message sequence numbers for a channel*

You can reset message sequence numbers for a channel using MQSC commands or using the operations and control panels.

To reset channel sequence numbers using the MQSC commands, use RESET CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	5 (Perform)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.USE
Disposition	The disposition of the channel object.

The Perform a Channel Function panel is displayed (see Figure 110 on page 990).

Select Function type 1 (reset).

Select the disposition of the channel for which the reset is to be done and on which queue manager it is to be done.

The **sequence number** field is initially set to one. Change this value if you want, and press enter.

z/OS *Resolving in-doubt messages on a channel*

You can resolve in-doubt messages on a channel using MQSC commands or using the operations and control panels.

To resolve in-doubt messages on a channel using the MQSC commands, use RESOLVE CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	5 (Perform)
Object type	SENDER, SERVER, or CHANNEL
Name	CHANNEL.TO.USE
Disposition	The disposition of the object.

The Perform a Channel Function panel is displayed (see Figure 110 on page 990).

Select Function type 3 or 4 (resolve with commit or backout). (See “Manejo de canales pendientes” on page 252 for more information.)

Select the disposition of the channel for which resolution is to be done and which queue manager it is to be done on. Press enter.

z/OS *Stopping a channel*

You can stop a channel using MQSC commands or using the operations and control panels.

To stop a channel using the MQSC commands, use STOP CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	7 (Stop)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.USE
Disposition	The disposition of the object.

The Stop a Channel panel is displayed. The text following the panel explains how to use the panel:

```
Stop a Channel

Complete fields, then press Enter to stop channel.


Channel name . . . . . : CHANNEL.TO.USE
Channel type . . . . . : SENDER
Description . . . . . : Description of CHANNEL.TO.USE


Disposition . . . . . P   P=Private on MQ25
A=Shared on any queue manager

Stop mode . . . . . 1   1. Quiesce  2. Force
Stop status . . . . . 1   1. Stopped  2. Inactive

Queue manager . . . . . ________________________________________________
Connection name . . . . . ________________________________________________



Command ===> ____________________________________________________________
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

Figure 111. Stopping a channel

Select the disposition of the channel for which the stop is to be done and on which queue manager it is to be stopped.

Choose the stop mode that you require:

Quiesce
The channel stops when the current message is completed and the batch is then ended, even if the batch size value has not been reached and there are messages already waiting on the transmission queue. No new batches are started. This mode is the default.

Force
The channel stops immediately. If a batch of messages is in progress, an 'in-doubt' situation can result.

Choose the queue manager and connection name for the channel you want to stop.

Choose the status that you require:

Stopped
The channel is not restarted automatically, and must be restarted manually. This mode is the default if no queue manager or connection name is specified. If a name is specified, it is not allowed.

Inactive
The channel is restarted automatically when required. This mode is the default if a queue manager or connection name is specified.

Press enter to stop the channel.

See "Detención y desactivación temporal de canales" on page 250 for more information. For information about restarting stopped channels, see "Reinicio de canales detenidos" on page 251.

Note: If a shared channel is in a retry state and the channel initiator on which it was started is not running, a STOP request for the channel is issued on the queue manager where the command was entered.

z/OS *Displaying channel status*

You can display channel status by using MQSC commands, or by using the operations and control panels.

To display the status of a channel or a set of channels using the MQSC commands, use DISPLAY CHSTATUS.

Note: Displaying channel status information can take some time if you have lots of channels.

Using the operations and control panels on the List Channel panel (see Figure 104 on page 982), a summary of the channel status is shown for each channel as follows:

INACTIVE	No connections are active
status	One connection is active
nnn status	More than one connection is current and all current connections have the same status
nnn CURRENT	More than one connection is current and the current connections do not all have the same status
Blank	IBM MQ is unable to determine how many connections are active (for example, because the channel initiator is not running) **Note:** For channel objects with the disposition GROUP, no status is displayed.

where *nnn* is the number of active connections, and *status* is one of the following:

INIT	INITIALIZING
BIND	BINDING
START	STARTING

RUN	RUNNING
STOP	STOPPING or STOPPED
RETRY	RETRYING
REQST	REQUESTING

To display more information about the channel status, press the Status key (F11) on the List Channel or the Display, or Alter channel panels to display the List Channels - Current Status panel (see Figure 112 on page 993).

```
List Channels - Current Status - MQ25      Row 1 of 16

Type action codes, then press Enter. Press F11 to display saved status.
1=Display current status

Channel name     Connection name                State
Start time      Messages Last message time  Type   Disposition
<> *                           CHANNEL  ALL   MQ25

_ RMA0.CIRCUIT.ACL.F  RMA1                        STOP
_  2005-03-21 10.22.36 557735  2005-03-24 09.51.11 SENDER  PRIVATE MQ25
_ RMA0.CIRCUIT.ACL.N  RMA1
_  2005-03-21 10.23.09 378675  2005-03-24 09.51.10 SENDER  PRIVATE MQ25
_ RMA0.CIRCUIT.CL.F  RMA2
_  2005-03-24 01.12.51 45544   2005-03-24 09.51.08 SENDER  PRIVATE MQ25
_ RMA0.CIRCUIT.CL.N  RMA2
_  2005-03-24 01.13.55 45560   2005-03-24 09.51.11 SENDER  PRIVATE MQ25
_ RMA1.CIRCUIT.CL.F  RMA1
_  2005-03-21 10.24.12 360757  2005-03-24 09.51.11 RECEIVER PRIVATE MQ25
_ RMA1.CIRCUIT.CL.N  RMA1
_  2005-03-21 10.23.40 302870  2005-03-24 09.51.09 RECEIVER PRIVATE MQ25
******** End of list ********
Command ===> ____________________________________________________________
F1=Help   F2=Split   F3=Exit   F4=Filter  F5=Refresh  F7=Bkwd
F8=Fwd    F9=SwapNext F10=Messages F11=Saved  F12=Cancel
```

Figure 112. Listing channel connections

The values for status are as follows:

INIT	INITIALIZING
BIND	BINDING
START	STARTING
RUN	RUNNING
STOP	STOPPING or STOPPED
RETRY	RETRYING
REQST	REQUESTING
DOUBT	STOPPED and INDOUBT(YES)

See “Estados de un canal” on page 242 for more information.

You can press F11 to see a similar list of channel connections with saved status; press F11 to get back to the current list. The saved status does not apply until at least one batch of messages has been transmitted on the channel.

Use action code 1 or a slash (/) to select a connection and press enter. The Display Channel Connection Current Status panels are displayed.

z/OS *Displaying cluster channels*

You can display cluster channels using MQSC commands or using the operations and control panels.

To display all the cluster channels that have been defined (explicitly or using auto-definition), use the MQSC command, DISPLAY CLUSQMGR.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	1 (List or Display)
Object type	CLUSCHL
Name	*

You are presented with a panel like figure Figure 113 on page 994, in which the information for each cluster channel occupies three lines, and includes its channel, cluster, and queue manager names. For cluster-sender channels, the overall state is shown.

```
List Cluster queue manager Channels - MQ25    Row 1 of 9

Type action codes, then press Enter. Press F11 to display connection status.
1=Display  5=Perform  6=Start  7=Stop

Channel name     Connection name                State
Type     Cluster name                   Suspended
Cluster queue manager name            Disposition
<>   *                          -    MQ25
_ TO.MQ90.T      HURSLEY.MACH90.COM(1590)
_  CLUSRCVR   VJH01T                        N
_   MQ90                        -    MQ25
_ TO.MQ95.T      HURSLEY.MACH95.COM(1595)             RUN
_  CLUSSDRA   VJH01T                        N
_   MQ95                        -    MQ25
_ TO.MQ96.T      HURSLEY.MACH96.COM(1596)             RUN
_  CLUSSDRB   VJH01T                        N
_   MQ96                        -    MQ25
******** End of list ********


Command ===> ____________________________________________________________________
F1=Help   F2=Split   F3=Exit   F4=Filter  F5=Refresh  F7=Bkwd
F8=Fwd    F9=SwapNext F10=Messages F11=Status  F12=Cancel
```

Figure 113. Listing cluster channels

To display full information about one or more channels, type Action code 1 against their names and press enter. Use Action codes 5, 6, or 7 to perform functions (such as ping, resolve, and reset), and start or stop a cluster channel.

To display more information about the channel status, press the Status key (F11).

z/OS *Preparing IBM MQ for z/OS to use the zEnterprise Data Compression Express facility*

The zEnterprise® Data Compression (zEDC) Express facility is available for certain models of IBM Z machines, starting from IBM zEC12 GA2, using a minimum z/OS level of z/OS 2.1.

See zEnterprise Data Compression (zEDC) for further information.

Prerequisites

For IBM z15 and later, the zEnterprise Data Compression (zEDC) Express facility was moved from an optional feature in the PCIe I/O drawer of the hardware system to be on-chip as the Integrated

Accelerator for zEDC. With this change, the configuration prerequisites are updated and are dependent on your hardware system.

IBM z15 or later

Apply one of the following PTFs, according to your level of z/OS:

- z/OS 2.4: UJ00636
- z/OS 2.3: UJ00635
- z/OS 2.2: UJ00638
- z/OS 2.1: UJ00639

There are no hardware requirements for z15 or later systems. The Integrated Accelerator for zEDC solution in these systems provides built-in data acceleration, so a separate adapter is no longer required.

IBM zEC12 GA2 to IBM z14

Your system must also have the following requirements:

- A zEDC Express® adapter, installed in the PCIe I/O drawers of the hardware system.
- The zEDC software capability (an optional, paid-for feature) must be enabled in an IFAPRDxx parmlib member.

Procedure

IBM zEC12 GA2 to IBM z14

Ensure that the channel initiator user ID has READ authority to the FPZ.ACCELERATOR.COMPRESSION profile in the RACF FACILITY CLASS, or the equivalent in the external security manager (ESM) that your enterprise uses.

Attention: Not required for IBM z15 or later.

IBM zEnterprise zEC12 GA2 or later

Configure the channel with COMPMSG(ZLIBFAST) at both the sending and receiving ends. Once configured, zlib compression is used to compress and decompress messages flowing across the channel.

Compression is performed in the zEDC when the size of the data to be compressed is above the minimum threshold. The threshold is dependent upon the IBM z hardware being used

- IBM zEC12 GA2 to IBM z14 has a minimum threshold of 4KB
- IBM z15 or later has a minimum threshold of 1KB

For messages below the threshold size, compression or inflation is performed in the software.

z/OS Setting up communication for z/OS

When a distributed-queuing management channel is started, it tries to use the connection specified in the channel definition. To succeed, it is necessary for the connection to be defined and available. This section explains how to define a connection.

DQM is a remote queuing facility for IBM MQ. It provides channel control programs for the queue manager that form the interface to communication links. These links are controllable by the system operator. The channel definitions held by distributed queuing management use these connections.

Choose from one of the two forms of communication protocol that can be used for z/OS:

- “Defining a TCP connection on z/OS” on page 996
- “Defining an LU6.2 connection for z/OS using APPC/MVS” on page 999

MQ Adv. CD Un canal de mensajes que utiliza TCP/IP puede apuntar a un IBM Aspera faspio Gateway, que proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de la red. Un gestor de colas que se ejecuta en cualquier plataforma autorizada puede conectarse a través de un Aspera gateway. La propia pasarela se despliega en Red Hat o Ubuntu Linux, o Windows. Consulte Definición de una conexión de Aspera gateway en Linux o Windows.

Each channel definition must specify only one protocol as the transmission protocol (Transport Type) attribute. A queue manager can use more than one protocol to communicate.

You might also find it helpful to refer to Example configuration - IBM MQ for z/OS . If you are using queue sharing groups, see “Setting up communication for IBM MQ for z/OS using queue sharing groups” on page 1004.

Related concepts

“Using the panels and the commands” on page 981
You can use the MQSC commands, the PCF commands, or the operations and control panels to manage DQM.

“Setting up IBM MQ for z/OS” on page 901
Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

“Monitoring and controlling channels on z/OS” on page 979
Use the DQM commands and panels to create, monitor, and control the channels to remote queue managers.

“Preparing IBM MQ for z/OS for DQM with queue sharing groups” on page 1000
Use the instructions in this section to configure distributed queuing with queue sharing groups on IBM MQ for z/OS.

“Setting up communication for IBM MQ for z/OS using queue sharing groups” on page 1004
When a distributed-queuing management channel is started, it attempts to use the connection specified in the channel definition. For this attempt to succeed, it is necessary for the connection to be defined and available.

Related tasks

“Configuración de la gestión de colas distribuidas” on page 210
En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

“Setting up communications with other queue managers on z/OS” on page 975
This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

z/OS *Defining a TCP connection on z/OS*

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

The TCP address space name must be specified in the TCP system parameters data set, *tcpip*.TCPIP.DATA. In the data set, a "TCPIPJOBNAME *TCPIP_proc*" statement must be included.

If you are using a firewall, you need to configure `allow` connections from the channel initiator to the addresses in the channels, and from the remote connections into the queue manager.

Typically the definition for a firewall configures the sending IP address and port to the destination IP address and port:

- A z/OS image can have more than one host name, and you might need to configure the firewall with multiple host addresses as the source address.

  You can use the NETSTAT HOME command to display these names and addresses.
- A channel initiator can have multiple listeners on different ports, so you need to configure these ports.
- If you are using a shared port for a queue sharing group you must configure the shared port as well.

The channel initiator address space must have authority to read the data set. The following techniques can be used to access your TCPIP.DATA data set, depending on which TCP/IP product and interface you are using:

- Environment variable, RESOLVER_CONFIG
- /etc/resolv.conf on the file system
- //SYSTCPD DD statement
- //SYSTCPDD DD statement
- *jobname/userid*.TCPIP.DATA
- SYS1.TCPPARMS(TCPDATA)
- *zapname*.TCPIP.DATA

You must also be careful to specify the high-level qualifier for TCP/IP correctly.

You need a suitably configured Domain Name System (DNS) server, capable of both Name to IP address translation and IP address to Name translation.

Note: Some changes to the resolver configuration require a recycle of applications using it, for example, IBM MQ.

For more information, see the following:

- Base TCP/IP system
- z/OS UNIX System Services.

Each TCP channel when started uses TCP resources; you might need to adjust the following parameters in your PROFILE.TCPIP configuration data set:

ACBPOOLSIZE
Add one per started TCP channel, plus one

CCBPOOLSIZE
Add one per started TCP channel, plus one per DQM dispatcher, plus one

DATABUFFERPOOLSIZE
Add two per started TCP channel, plus one

MAXFILEPROC
Controls how many channels each dispatcher in the channel initiator can handle.

This parameter is specified in the BPXPRMxx member of SYSI.PARMLIB. Ensure that you specify a value large enough for your needs.

By default, the channel initiator is only capable of binding to IP addresses associated with the stack named in the TCPNAME queue manager attribute. To allow the channel initiator to communicate using additional TCP/IP stacks on the system, change the TCPSTACK queue manager attribute to MULTIPLE.

Un canal de mensajes que utiliza TCP/IP puede apuntar a un IBM Aspera faspio Gateway, que proporciona un túnel TCP/IP rápido que puede aumentar significativamente el rendimiento de la red. Un gestor de colas que se ejecuta en cualquier plataforma autorizada puede conectarse a través de un Aspera gateway. La propia pasarela se despliega en Red Hat o Ubuntu Linux, o Windows. Consulte Definición de una conexión de Aspera gateway en Linux o Windows.

Related concepts

"Sending end" on page 998
At the sending end of the TCP/IP connection, there are a number of settings to configure.

"Receiving on TCP" on page 998
At the receiving end of the TCP/IP connection, there are a number of settings to configure.

"Using the TCP listener backlog option on z/OS" on page 998
When receiving on TCP/IP, a maximum number of outstanding connection requests is set. These outstanding requests can be considered a *backlog* of requests waiting on the TCP/IP port for the listener to accept the request.

z/OS *Sending end*

At the sending end of the TCP/IP connection, there are a number of settings to configure.

The connection name (CONNAME) field in the channel definition must be set to either the host name (for example MVSHUR1) or the TCP network address of the target. The TCP network address can be in IPv4 dotted decimal form (for example 127.0.0.1) or IPv6 hexadecimal form (for example 2001:DB8:0:0:0:0:0:0). If the connection name is a host name, a TCP name server is required to convert the host name into a TCP host address. (This requirement is a function of TCP, not IBM MQ.)

On the initiating end of a connection (sender, requester, and server channel types) it is possible to provide an optional port number for the connection, for example:

Connection name
192.0.2.0(1555)

In this case the initiating end attempts to connect to a receiving program listening on port 1555.

Note: The default port number of 1414 is used if an optional port number is not specified.

The channel initiator can use any TCP/IP stack which is active and available. By default, the channel initiator binds its outbound channels to the default IP address for the TCP/IP stack named in the TCPNAME queue manager attribute. To connect through a different stack, you need to specify either the host name or IP address of the stack in the LOCLADDR attribute of the channel.

z/OS *Receiving on TCP*

At the receiving end of the TCP/IP connection, there are a number of settings to configure.

Receiving channel programs are started in response to a startup request from the sending channel. To do so, a listener program has to be started to detect incoming network requests and start the associated channel. You start this listener program with the START LISTENER command, or using the operations and control panels.

By default:

- The TCP Listener program uses port 1414 and listens on all addresses available to your TCP stack.
- TCP/IP listeners can bind only to addresses associated with the TCP/IP stack named in the TCPNAME queue manager attribute.

To start listeners for other addresses, or all available TCP stacks, set your TCPSTACK queue manager attribute to 'MULTIPLE'.

You can start your TCP listener program to listen only on a specific address or host name by specifying IPADDR in the START LISTENER command. For more information, see Listeners.

z/OS *Using the TCP listener backlog option on z/OS*

When receiving on TCP/IP, a maximum number of outstanding connection requests is set. These outstanding requests can be considered a *backlog* of requests waiting on the TCP/IP port for the listener to accept the request.

The default listener backlog value on z/OS is 10000. If the backlog reaches this values, the TCP/IP connection is rejected and the channel is not able to start.

For MCA channels, this results in the channel going into a RETRY state and retrying the connection at a later time.

For client connections, the client receives an MQRC_Q_MGR_NOT_AVAILABLE reason code from MQCONN and can retry the connection at a later time.

Related concepts

“Utilización de la opción de proceso de escucha TCP en IBM MQ for Multiplatforms” on page 284
En TCP, las conexiones se tratan de forma incompleta a menos que tenga lugar un reconocimiento entre el servidor y el cliente. Estas conexiones se llaman solicitudes de conexión pendientes. Se establece un valor máximo para estas solicitudes de conexión pendientes y se puede considerar una reserva de solicitudes en espera del puerto TCP para que el escucha acepte la solicitud.

z/OS *Defining an LU6.2 connection for z/OS using APPC/MVS*

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

APPC/MVS setup

Each instance of the channel initiator must have the name of the LU that it is to use defined to APPC/MVS, in the APPCPMxx member of SYS1.PARMLIB, as in the following example:

```
LUADD ACBNAME( luname ) NOSCHED TPDATA(CSQ.APPCTP)
```

luname is the name of the logical unit to be used. NOSCHED is required; TPDATA is not used. No additions are necessary to the ASCHPMxx member, or to the APPC/MVS TP profile data set.

The side information data set must be extended to define the connections used by DQM. See the supplied sample CSQ4SIDE for details of how to do this using the APPC utility program ATBSDFMU. For details of the TPNAME values to use, see the following table for information:

Table 67. Settings on the local z/OS system for a remote queue manager platform

Remote platform	TPNAME
z/OS or MVS	The same as TPNAME in the corresponding side information about the remote queue manager.
IBM i	The same as the compare value in the routing entry on the IBM i system.
AIX and Linux systems	The same as TPNAME in the corresponding side information about the remote queue manager.
Windows	As specified in the Windows Run Listener command, or the invokable Transaction Program that was defined using TpSetup on Windows.

If you have more than one queue manager on the same machine, ensure that the TPnames in the channel definitions are unique.

In an environment where the queue manager is communicating using APPC with a queue manager on the same or another z/OS system, ensure that either the VTAM definition for the communicating LU specifies SECACPT(ALREADYV), or that there is a RACF APPCLU profile for the connection between LUs, which specifies CONVSEC(ALREADYV).

The z/OS command VARY ACTIVE must be issued against both base and listener LUs before attempting to start either inbound or outbound communications.

Attention: In addition to the APPC setup, you must issue the following command:

```
ALTER QMGR LUNAME(luname)
```

and restart the channel initiator.

See LUNAME for further information.

Related concepts

“Connecting to LU 6.2” on page 999
To connect to LU 6.2, there are a number of settings to configure.

“Receiving on LU 6.2” on page 1000
To receive on LU 6.2, there are a number of settings to configure.

z/OS *Connecting to LU 6.2*

To connect to LU 6.2, there are a number of settings to configure.

The connection name (CONNAME) field in the channel definition must be set to the symbolic destination name, as specified in the side information data set for APPC/MVS.

The LU name to use (defined to APPC/MVS as described previously) must also be specified in the channel initiator parameters. It must be set to the same LU that is used for receiving by the listener.

The channel initiator uses the "SECURITY(SAME)" APPC/MVS option, so it is the user ID of the channel initiator address space that is used for outbound transmissions, and is presented to the receiver.

z/OS *Receiving on LU 6.2*

To receive on LU 6.2, there are a number of settings to configure.

Receiving MCAs are started in response to a startup request from the sending channel. To do so, a listener program has to be started to detect incoming network requests and start the associated channel. The listener program is an APPC/MVS server. You start it with the START LISTENER command, or using the operations and control panels. You must specify the LU name to use with a symbolic destination name defined in the side information data set. The local LU so identified must be the same as the one used for outbound transmissions, as set in the channel initiator parameters.

z/OS Preparing IBM MQ for z/OS for DQM with queue sharing groups

Use the instructions in this section to configure distributed queuing with queue sharing groups on IBM MQ for z/OS.

For an example configuration using queue sharing groups, see Example configuration - IBM MQ for z/OS using queue sharing groups. For a message channel planning example using queue sharing groups, see Message channel planning example for z/OS using queue sharing groups.

You need to create and configure the following components to enable distributed queuing with queue sharing groups:

- LU 6.2 and TCP/IP listeners
- Transmission queues and triggering
- Message channel agents
- Synchronization queue

After you have created the components you need to set up the communication, see “Setting up communication for IBM MQ for z/OS using queue sharing groups” on page 1004.

For information about how to monitor and control channels when using queue sharing groups, see “Monitoring and controlling channels on z/OS” on page 979.

See the following sections for queue sharing group concepts and benefits.

Class of service

A shared queue is a type of local queue that offers a different class of service. Messages on a shared queue are stored in a coupling facility (CF), which allows them to be accessed by all queue managers in the queue sharing group. A message on a shared queue must be a message of length no more than 100 MB.

Generic interface

A queue sharing group has a generic interface that allows the network to view the group as a single entity. This view is achieved by having a single generic address that can be used to connect to any queue manager within the group.

Each queue manager in the queue sharing group listens for inbound session requests on an address that is logically related to the generic address. For more information see “LU 6.2 and TCP/IP listeners for queue sharing groups” on page 1002.

Load-balanced channel start

A shared transmission queue can be serviced by an outbound channel running on any channel initiator in the queue sharing group. Load-balanced channel start determines where a start channel command is targeted. An appropriate channel initiator is chosen that has access to the necessary communications subsystem. For example, a channel defined with TRPTYPE(LU6.2) cannot be started on a channel initiator that only has access to a TCP/IP subsystem.

The choice of channel initiator is dependent on the channel load and the headroom of the channel initiator. The channel load is the number of active channels as a percentage of the maximum number of active channels allowed as defined in the channel initiator parameters. The headroom is the difference between the number of active channels and the maximum number allowed.

Inbound shared channels can be load-balanced across the queue sharing group by use of a generic address, as described in “LU 6.2 and TCP/IP listeners for queue sharing groups” on page 1002.

Shared channel recovery

The following table shows the types of shared-channel failure and how each type is handled.

Type of failure:	**What happens:**
Channel initiator communications subsystem failure	The channels dependent on the communications subsystem enter channel retry, and are restarted on an appropriate queue sharing group channel initiator by a load-balanced start command.
Channel initiator failure	The channel initiator fails, but the associated queue manager remains active. The queue manager monitors the failure and initiates recovery processing.
Queue manager failure	The queue manager fails (failing the associated channel initiator). Other queue managers in the queue sharing group monitor the event and initiate peer recovery.
Shared status failure	Channel state information is stored in Db2, so a loss of connectivity to Db2 becomes a failure when a channel state change occurs. Running channels can carry on running without access to these resources. On a failed access to Db2, the channel enters retry.

Shared channel recovery processing on behalf of a failed system requires connectivity to Db2 to be available on the system managing the recovery to retrieve the shared channel status.

Client channels

Client connection channels can benefit from the high availability of messages in queue sharing groups that are connected to the generic interface instead of being connected to a specific queue manager. For more information, see Client connection channels.

Related concepts

Shared queues and queue sharing groups

“Setting up IBM MQ for z/OS” on page 901
Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

“Clusters and queue sharing groups” on page 1003
You can make your shared queue available to a cluster in a single definition. To do so you specify the name of the cluster when you define the shared queue.

“Channels and serialization” on page 1004
During shared queue peer recovery, message channel agents that process messages on shared queues serialize their access to the queues.

Intra-group queuing

Related tasks

"Configuración de la gestión de colas distribuidas" on page 210
En esta sección se proporciona información más detallada sobre la intercomunicación entre instalaciones de IBM MQ, incluyendo la definición de cola, la definición de canal, el mecanismo de desencadenamiento y los procedimientos de punto de sincronización

"Setting up communications with other queue managers on z/OS" on page 975
This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

z/OS *LU 6.2 and TCP/IP listeners for queue sharing groups*

The group LU 6.2 and TCP/IP listeners listen on an address that is logically connected to the generic address.

For the LU 6.2 listener, the specified LUGROUP is mapped to the VTAM generic resource associated with the queue sharing group. For an example of setting up this technology, see "Defining an LU6.2 connection for z/OS using APPC/MVS" on page 999.

For the TCP/IP listener, the specified port can be connected to the generic address in one of the following ways:

- For a front-end router such as the IBM Network Dispatcher, inbound connect requests are forwarded from the router to the members of the queue sharing group.
- For TCP/IP Sysplex Distributor, each listener that is running and is listening on a particular address that is set up as a Distributed DVIPA is allocated a proportion of the incoming requests. For an example of setting up this technology, see Using Sysplex Distributor

z/OS *Transmission queues and triggering for queue sharing groups*

A shared transmission queue is used to store messages before they are moved from the queue sharing group to the destination.

It is a shared queue and it is accessible to all queue managers in the queue sharing group.

Triggering

A triggered shared queue can generate more than one trigger message for a satisfied trigger condition. There is one trigger message generated for each local initiation queue defined on a queue manager in the queue sharing group associated with the triggered shared queue.

For distributed queuing, each channel initiator receives a trigger message for a satisfied shared transmission queue trigger condition. However, only one channel initiator actually processes the triggered start, and the others fail safely. The triggered channel is then started with a load balanced start (see "Preparing IBM MQ for z/OS for DQM with queue sharing groups" on page 1000) that is triggered to start channel `QSG.TO.QM2`. To create a shared transmission queue, use the IBM MQ commands (MQSC) as shown in the following example:

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') +
USAGE(XMITQ) QSGDISP(SHARED) +
CFSTRUCT(APPLICATION1) INITQ(SYSTEM.CHANNEL.INITQ) +
TRIGGER TRIGDATA(QSG.TO.QM2)
```

Note: If a shared queue is setup for triggering and connection to the Coupling Facility hosting the shared queue is lost, a trigger event might be generated and a message put to the initiation queue. This can happen even when no message was put to the original shared queue setup for triggering. This is caused by the over-indication of bits by the IXLVECTR macro as documented in The List Notification Vector.

z/OS *Message channel agents for queue sharing groups*

A channel can only be started on a channel initiator if it has access to a channel definition for a channel with that name.

A message channel agent is an IBM MQ program that controls the sending and receiving of messages. Message channel agents move messages from one queue manager to another; there is one message channel agent at each end of a channel.

A channel definition can be defined to be private to a queue manager or stored on the shared repository and available anywhere (a group definition). This means that a group defined channel is available on any channel initiator in the queue sharing group.

Note: The private copy of the group definition can be changed or deleted.

To create group channel definitions, use the IBM MQ commands (MQSC) as shown in the following examples:

```
DEFINE CHL(QSG.TO.QM2) CHLTYPE(SDR) +
TRPTYPE(TCP) CONNAME(QM2.MACH.IBM.COM) +
XMITQ(QM2) QSGDISP(GROUP)
```

```
DEFINE CHL(QM2.TO.QSG) CHLTYPE(RCVR) TRPTYPE(TCP) +
QSGDISP(GROUP)
```

There are two perspectives from which to look at the message channel agents used for distributed queuing with queue sharing groups:

Inbound

An inbound channel is a shared channel if it is connected to the queue manager through the group listener. It is connected either through the generic interface to the queue sharing group, then directed to a queue manager within the group, or targeted at the group port of a specific queue manager or the luname used by the group listener.

Outbound

An outbound channel is a shared channel if it moves messages from a shared transmission queue. In the example commands, sender channel `QSG.TO.QM2` is a shared channel because its transmission queue, QM2 is defined with QSGDISP(SHARED).

z/OS *Synchronization queue for queue sharing groups*

Shared channels have their own shared synchronization queue called SYSTEM.QSG.CHANNEL.SYNCQ.

This synchronization queue is accessible to any member of the queue sharing group. (Private channels continue to use the private synchronization queue. See “Defining IBM MQ objects on z/OS” on page 978). This means that the channel can be restarted on a different queue manager and channel initiator instance within the queue sharing group in the event of failure of the communications subsystem, channel initiator, or queue manager. For further information, see “Preparing IBM MQ for z/OS for DQM with queue sharing groups” on page 1000.

DQM with queue sharing groups requires that a shared queue is available with the name SYSTEM.QSG.CHANNEL.SYNCQ. This queue must be available so that a group listener can successfully start.

If a group listener fails because the queue was not available, the queue can be defined and the listener can be restarted without recycling the channel initiator. The non-shared channels are not affected.

Make sure that you define this queue using INDXTYPE(MSGID). This definition improves the speed at which messages on the queue can be accessed.

z/OS *Clusters and queue sharing groups*

You can make your shared queue available to a cluster in a single definition. To do so you specify the name of the cluster when you define the shared queue.

Users in the network see the shared queue as being hosted by each queue manager within the queue sharing group. (The shared queue is not advertised as being hosted by the queue sharing group). Clients can start sessions with all members of the queue sharing group to put messages to the same shared queue.

For more information, see “Configuración de un clúster de gestores de colas” on page 309.

z/OS *Channels and serialization*

During shared queue peer recovery, message channel agents that process messages on shared queues serialize their access to the queues.

If a queue manager in a queue sharing group fails while a message channel agent is dealing with uncommitted messages on one or more shared queues, the channel and the associated channel initiator will end, and shared queue peer recovery will take place for the queue manager.

Because shared queue peer recovery is an asynchronous activity, peer channel recovery might try to simultaneously restart the channel in another part of the queue sharing group before shared queue peer recovery is complete. If this event happens, committed messages might be processed ahead of the messages still being recovered. To ensure that messages are not processed out of sequence in this way, message channel agents that process messages on shared queues serialize their access to these queues.

An attempt to start a channel for which shared queue peer recovery is still in progress might result in a failure. An error message indicating that recovery is in progress is issued, and the channel is put into retry state. Once queue manager peer recovery is complete, the channel can restart at the time of the next retry.

An attempt to RESOLVE, PING, or DELETE a channel can fail for the same reason.

z/OS *Setting up communication for IBM MQ for z/OS using queue sharing groups*

When a distributed-queuing management channel is started, it attempts to use the connection specified in the channel definition. For this attempt to succeed, it is necessary for the connection to be defined and available.

Choose from one of the two forms of communication protocol that can be used:

- TCP
- LU 6.2 through APPC/MVS

You might find it useful to refer to Example configuration - IBM MQ for z/OS using queue sharing groups.

z/OS *Defining a TCP connection for queue sharing groups*

To define a TCP connection for a queue sharing group, certain attributes on the sending and receiving end must be configured.

For information about setting up your TCP, see “Defining a TCP connection on z/OS” on page 996.

Sending end

The connection name (CONNAME) field in the channel definition to connect to your queue sharing group must be set to the generic interface of your queue sharing group (see Queue sharing groups). For more details, refer to Using Sysplex Distributor.

Receiving on TCP using a queue sharing group

Receiving shared channel programs are started in response to a startup request from the sending channel. To do so, a listener must be started to detect incoming network requests and start the associated channel. You start this listener program with the START LISTENER command, using the inbound disposition of the group, or using the operations and control panels.

All group listeners in the queue sharing group must be listening on the same port. If you have more than one channel initiator running on a single MVS image you can define virtual IP addresses and start your

TCP listener program to only listen on a specific address or host name by specifying IPADDR in the START LISTENER command. (For more information, see START LISTENER.)

z/OS *Defining an LU 6.2 connection on z/OS*

To define an LU 6.2 connection for a queue sharing group, certain attributes on the sending and receiving end must be configured.

For information about setting up APPC/MVS, see Setting up communication for z/OS .

Connecting to APPC/MVS (LU 6.2)

The connection name (CONNAME) field in the channel definition to connect to your queue sharing group must be set to the symbolic destination name, as specified in the side information data set for APPC/MVS. The partner LU specified in this symbolic destination must be the generic resource name. For more details, see Defining yourself to the network using generic resources.

Receiving on LU 6.2 using a generic interface

Receiving shared MCAs are started in response to a startup request from the sending channel. To do so, a group listener program must be started to detect incoming network requests and start the associated channel. The listener program is an APPC/MVS server. You start it with the START LISTENER command, using an inbound disposition group, or using the operations and control panels. You must specify the LU name to use a symbolic destination name defined in the side information data set. For more details, see Defining yourself to the network using generic resources.

z/OS Using IBM MQ with IMS

The IBM MQ -IMS adapter, and the IBM MQ - IMS bridge are the two components which allow IBM MQ to interact with IMS.

To configure IBM MQ and IMS to work together, you must complete the following tasks:

- “Setting up the IMS adapter” on page 1005
- “Setting up the IMS bridge” on page 1012

Related concepts

IBM MQ and IMS

“Using IBM MQ with CICS” on page 1013
To use IBM MQ with CICS, you must configure the IBM MQ CICS adapter and, optionally, the IBM MQ CICS bridge components.

“Using OTMA exits in IMS” on page 1015
Use this topic if you want to use IMS Open Transaction Manager Access exits with IBM MQ for z/OS.

IMS and IMS bridge applications on IBM MQ for z/OS

Related tasks

“Configuring queue managers on z/OS” on page 896
Use these instructions to configure queue managers on IBM MQ for z/OS.

Related reference

“Upgrading and applying service to Language Environment or z/OS Callable Services” on page 1013
The actions you must take vary according to whether you use CALLLIBS or LINK, and your version of SMP/E.

z/OS Setting up the IMS adapter

To use IBM MQ within IMS requires the IBM MQ - IMS adapter (generally referred to as the IMS adapter).

This topic tells you how to make the IMS adapter available to your IMS subsystem. If you are not familiar with tailoring an IMS subsystem, see the IMS documentation .

To make the IMS adapter available to IMS applications, follow these steps:

1. Define IBM MQ to IMS as an external subsystem using the IMS external subsystem attach facility (ESAF).

   See “Defining IBM MQ to IMS” on page 1007.

2. Include the IBM MQ load library thlqual.SCSQAUTH in the JOBLIB or STEPLIB concatenation in the JCL for your IMS control region and for any dependent region that connects to IBM MQ (if it is not in the LPA or link list). If your JOBLIB or STEPLIB is not authorized, also include it in the DFSESL concatenation after the library containing the IMS modules (usually IMS RESLIB).

   Also include thlqual.SCSQANLx (where x is the language letter).

   If DFSESL is present, then SCSQAUTH and SCSQANLx need to be included in the concatenation or added to LNKLIST. Adding to the STEPLIB or JOBLIB concatenation in the JCL is not sufficient.

3. Copy the IBM MQ assembler program CSQQDEFV from thlqual.SCSQASMS to a user library.

4. The supplied program, CSQQDEFV, contains one subsystem name CSQ1 identified as default with an IMS language interface token (LIT) of MQM1. You can retain this name for testing and installation verification.

   For production subsystems, you change the NAME=CSQ1 to your own subsystem name, or use CSQ1. You can add further subsystem definitions as required. See “Defining IBM MQ queue managers to the IMS adapter” on page 1010 for further information on LITs.

5. Assemble and link-edit the program to produce the CSQQDEFV load module. For the assembly, include the library thlqual.SCSQMACS in your SYSLIB concatenation; use the link-edit parameter RENT. This is shown in the sample JCL in thlqual.SCSQPROC(CSQ4DEFV).

6. Include the user library containing the module CSQQDEFV that you created in the JOBLIB or STEPLIB concatenation in the JCL for any dependent region that connects to IBM MQ. Put this library before the SCSQAUTH because SCSQAUTH has a default load module. If you do not do this, you will receive a user 3041 abend from IMS.

7. If the IMS adapter detects an unexpected IBM MQ error, it issues a z/OS SNAP dump to DD name CSQSNAP and issues reason code MQRC_UNEXPECTED_ERROR to the application. If the CSQSNAP DD statement was not in the IMS dependent region JCL, no dump is taken. If this happens, you could include the CSQSNAP DD statement in the JCL and rerun the application. However, because some problems might be intermittent, it is recommended that you include the CSQSNAP DD statement to capture the reason for failure at the time it occurs.

8. If you want to use dynamic IBM MQ calls (described in Dynamically calling the IBM MQ stub), build the dynamic stub, as shown in Figure 114 on page 1007.

9. If you want to use the IMS trigger monitor, define the IMS trigger monitor application CSQQTRMN, and perform PSBGEN and ACBGEN. See “Setting up the IMS trigger monitor” on page 1011.

10. If you are using RACF to protect resources in the OPERCMDS class, ensure that the userid associated with your IBM MQ queue manager address space has authority to issue the MODIFY command to any IMS system to which it might connect.

```
//DYNSTUB EXEC PGM=IEWL,PARM='RENT,REUS,MAP,XREF'
//SYSPRINT DD  SYSOUT=*
//ACSQMOD  DD  DISP=SHR,DSN=thlqual.SCSQLOAD
//IMSLIB   DD  DISP=SHR,DSN=ims.reslib
//SYSLMOD  DD  DISP=SHR,DSN=private.load[1]
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,1)
//SYSLIN   DD  *
  INCLUDE ACSQMOD(CSQQSTUB)
  INCLUDE IMSLIB(DFSLI000)
  ALIAS MQCONN,MQCONNX,MQDISC             MQI entry points
  ALIAS MQGET,MQPUT,MQPUT1                MQI entry points
  ALIAS MQOPEN,MQCLOSE                    MQI entry points
  ALIAS MQBACK,MQCMIT                     MQI entry points
  ALIAS CSQBBAK,CSQBCMT                   MQI entry points
  ALIAS MQINQ,MQSET                       MQI entry points
  ALIAS DFSPLI,PLITDLI                    IMS entry points
  ALIAS DFSCOBOL,CBLTDLI                  IMS entry points
  ALIAS DFSFOR,FORTDLI                    IMS entry points
  ALIAS DFSASM,ASMTDLI                    IMS entry points
  ALIAS DFSPASCL,PASTDLI                  IMS entry points
  ALIAS DFHEI01,DFHEI1                    IMS entry points
  ALIAS DFSAIBLI,AIBTDLI                  IMS entry points
  ALIAS DFSESS,DSNWLI,DSNHLI              IMS entry points
  ALIAS MQCRTMH,MQDLTMH,MQDLTMP           IMS entry points
  ALIAS MQINQMP,MQSETMP,MQMHBUF,MQBUFMH   IMS entry points
  MODE AMODE(31),RMODE(24)                Note RMODE setting
  NAME CSQQDYNS(R)
/*

[1]Specify the name of a library accessible to IMS applications that
want to make dynamic calls to IBM MQ.
```

Figure 114. Sample JCL to link-edit the dynamic call stub

Related concepts

IBM MQ and IMS

“Setting up the IMS bridge” on page 1012

The IBM MQ - IMS bridge is an optional component that enables IBM MQ to input and output to and from existing programs and transactions that are not IBM MQ-enabled.

IMS and IMS bridge applications on IBM MQ for z/OS

z/OS *Defining IBM MQ to IMS*

IBM MQ must be defined to the IMS control region, and to each dependent region accessing that IBM MQ queue manager. To do this, you must create a subsystem member (SSM) in the IMS.PROCLIB library, and identify the SSM to the applicable IMS regions.

Placing the subsystem member entry in IMS.PROCLIB

Each SSM entry in IMS.PROCLIB defines a connection from an IMS region to a different queue manager.

To name an SSM, concatenate the value (one to four alphanumeric characters) of the IMSID field of the IMS IMSCTRL macro with any name (one to four alphanumeric characters) defined by your site.

One SSM can be shared by all the IMS regions, or a specific member can be defined for each region. This member contains as many entries as there are connections to external subsystems. Each entry is an 80-character record.

Positional parameters

The fields in this entry are:

```
SSN,LIT,ESMT,RTT,REO,CRC
```

where:

SSN
Specifies the IBM MQ queue manager name. It is required, and must contain one through four characters.

LIT
Specifies the language interface token (LIT) supplied to IMS. This field is required, its value must match one in the CSQQDEFV module.

ESMT
Specifies the external subsystem module table (ESMT). This table specifies which attachment modules must be loaded by IMS. CSQQESMT is the required value for this field.

RTT
This option is not supported by IBM MQ.

REO
Specifies the region error option (REO) to be used if an IMS application references a non-operational external subsystem or if resources are unavailable at create thread time. This field is optional and contains a single character, which can be:

R
Passes a return code to the application, indicating that the request for IBM MQ services failed.

Q
Ends the application with an abend code U3051, backs out activity to the last commit point, does a PSTOP of the transaction, and requeues the input message. This option only applies when an IMS application tries to reference a non-operational external subsystem or if the resources are unavailable at create thread time.

IBM MQ completion and reason codes are returned to the application if the IBM MQ problem occurs while IBM MQ is processing the request; that is, after the adapter has passed the request on to IBM MQ.

A
Ends the application with an abend code of U3047 and discards the input message. This option only applies when an IMS application references a non-operational external subsystem or if the resources are unavailable at create thread time.

IBM MQ completion and reason codes are returned to the application if the IBM MQ problem occurs while IBM MQ is processing the request; that is, after the adapter has passed the request on to IBM MQ.

CRC
This option can be specified but is not used by IBM MQ.

Note: For full details of all positional parameters refer to How external subsystems are specified to IMS.

An example SSM entry is:

```
CSQ1,MQM1,CSQQESMT,,R,
```

where:

CSQ1 The default subsystem name as supplied with IBM MQ. You can change this to suit your installation.

MQM1	The default LIT as supplied in CSQQDEFV.
CSQQESMT	The external subsystem module name. You must use this value.
R	REO option.

Keyword parameters

IBM MQ parameters can be specified in keyword format. The SST parameter can have a value of either DB2 or MQ. Support for the MQ value was added in IMS 14. Use of MQ aids clarity, and the IMS subsystem command now includes the SST value, but otherwise does not have any significant effect. A value of DB2 can still be used if required. Other parameters are as described in Positional parameters, and shown in the following example:

```
SST=MQ,SSN=SYS3,LIT=MQM3,ESMT=CSQQESMT
```

where:

SYS3	The subsystem name
MQM3	The LIT as supplied in CSQQDEFV
CSQQESMT	The external subsystem module name

Specifying the SSM EXEC parameter

Specify the SSM EXEC parameter in the startup procedure of the IMS control region. This parameter specifies the one-character to four-character subsystem member name (SSM).

If you specify the SSM for the IMS control region, any dependent region running under the control region can attach to the IBM MQ queue manager named in the IMS.PROCLIB member specified by the SSM parameter. The IMS.PROCLIB member name is the IMS ID (IMSID= *xxxx*) concatenated with the one to four characters specified in the SSM EXEC parameter. The IMS ID is the IMSID parameter of the IMSCTRL generation macro.

IMS lets you define as many external subsystem connections as are required. More than one connection can be defined for different IBM MQ queue managers. All IBM MQ connections must be within the same z/OS system. For a dependent region, you can specify a dependent region SSM or use the one specified for the control region. You can specify different region error options (REOs) in the dependent region SSM and the control region SSM. Table 68 on page 1009 shows the different possibilities of SSM specifications.

Table 68. SSM specifications options

SSM for control region	SSM for dependent region	Action	Comments
No	No	None	No external subsystem can be connected.
No	Yes	None	No external subsystem can be connected.
Yes	No	Use the control region SSM	Applications scheduled in the region can access external subsystems identified in the control region SSM. Exits and control blocks for each attachment are loaded into the control region and the dependent region address spaces.
Yes	Yes (empty)	No SSM is used for the dependent region	Applications scheduled in this region can access DL/I databases only. Exits and control blocks for each attachment are loaded into the control region address space.

Table 68. SSM specifications options (continued)			
SSM for control region	**SSM for dependent region**	**Action**	**Comments**
Yes	Yes (not empty)	Check the dependent region SSM with the control region SSM	Applications scheduled in this region can access only external subsystems identified in both SSMs. Exits and control blocks for each attachment are loaded into the control region and the dependent region address spaces.

There is no specific parameter to control the maximum number of SSM specification possibilities.

Preloading the IMS adapter

The performance of the IMS adapter can be improved if it is preloaded by IMS. Preloading is controlled by the DFSMPLxx member of IMS.PROCLIB: see " IMS Administration Guide: System" for more information. The IBM MQ module names to specify are:

CSQACLST	CSQAMLST	CSQAPRH	CSQAVICM	CSQFSALM	CSQQDEFV
CSQQCONN	CSQQDISC	CSQQTERM	CSQQINIT	CSQQBACK	CSQQCMMT
CSQQESMT	CSQQPREP	CSQQTTHD	CSQQWAIT	CSQQNORM	CSQQSSOF
CSQQSSON	CSQFSTAB	CSQQRESV	CSQQSNOP	CSQQCMND	CSQQCVER
CSQQTMID	CSQQTRGI	CSQQCON2	CSQBPAPI	CSQBCRMH	CSQBAPPL

For more information on the use of IBM MQ classes for JMS, see Using IBM MQ classes for JMS in IMS.

Current releases of IMS support preloading IBM MQ modules from PDS-E format libraries in MPP, BMP, IFP, JMP and JBP regions only. Any other type of IMS region does not support preloading from PDS-E libraries. If preloading is required for any other type of region, then the IBM MQ modules that are provided must be copied to a PDS format library.

z/OS *Defining IBM MQ queue managers to the IMS adapter*

The names of the IBM MQ queue managers and their corresponding language interface tokens (LITs) must be defined in the queue manager definition table.

Use the supplied CSQQDEFX macro to create the CSQQDEFV load module. Figure 115 on page 1010 shows the syntax of this assembler macro.

```
CSQQDEFX TYPE=ENTRY|DEFAULT,NAME=qmgr-name,LIT=token
or
CSQQDEFX TYPE=END
```

Figure 115. CSQQDEFX macro syntax

Parameters

TYPE=ENTRY|DEFAULT
Specify either TYPE=ENTRY or TYPE=DEFAULT as follows:

TYPE=ENTRY
Specifies that a table entry describing an IBM MQ queue manager available to an IMS application is to be generated. If this is the first entry, the table header is also generated, including a CSQQDEFV CSECT statement.

TYPE=DEFAULT
As for TYPE=ENTRY. The queue manager specified is the default queue manager to be used when MQCONN or MQCONNX specifies a name that is all blanks. There must be only one such entry in the table.

NAME= *qmgr-name*
Specifies the name of the queue manager, as specified with **MQCONN** or **MQCONNX**.

LIT= token
Specifies the name of the language interface token (LIT) that IMS uses to identify the queue manager.

An MQCONN or MQCONNX call associates the *name* input parameter and the *hconn* output parameter with the name label and, therefore, the LIT in the CSQQDEFV entry. Further IBM MQ calls passing the *hconn* parameter use the LIT from the CSQQDEFV entry identified in the MQCONN or MQCONNX call to direct calls to the IBM MQ queue manager defined in the IMS SSM PROCLIB member with that same LIT.

In summary, the **name** parameter on the MQCONN or MQCONNX call identifies a LIT in CSQQDEFV and the same LIT in the SSM member identifies an IBM MQ queue manager. (For information about the MQCONN call, see MQCONN - Connect queue manager. For information about the MQCONNX call, see MQCONNX - Connect queue manager (extended).)

TYPE=END
Specifies that the table is complete. If this parameter is omitted, TYPE=ENTRY is assumed.

Using the CSQQDEFX macro

Figure 116 on page 1011 shows the general layout of a queue manager definition table.

```
CSQQDEFX NAME=subsystem1,LIT=token1
CSQQDEFX NAME=subsystem2,LIT=token2,TYPE=DEFAULT
CSQQDEFX NAME=subsystem3,LIT=token3
...
CSQQDEFX NAME=subsystemN,LIT=tokenN
CSQQDEFX TYPE=END
END
```

Figure 116. Layout of a queue manager definition table

z/OS *Setting up the IMS trigger monitor*

You can set up an IMS batch-oriented program to monitor an IBM MQ initiation queue.

Define the application to IMS using the model CSQQTAPL in the thlqual.SCSQPROC library (see Example transaction definition for CSQQTRMN).

Generate the PSB and ACB using the model CSQQTPSB in the thlqual.SCSQPROC library (see Example PSB definition for CSQQTRMN).

```
*  This is the application definition *
*  for the IMS Trigger Monitor BMP    *

APPLCTN PSB=CSQQTRMN,
PGMTYPE=BATCH,
SCHDTYP=PARALLEL
```

Figure 117. Example transaction definition for CSQQTRMN

```
PCB  TYPE=TP,      ALTPCB for transaction messages
MODIFY=YES,        To "triggered" IMS transaction
PCBNAME=CSQQTRMN
PCB  TYPE=TP,      ALTPCB for diagnostic messages
MODIFY=YES,        To LTERM specified or "MASTER"
PCBNAME=CSQQTRMG,
EXPRESS=YES
PSBGEN LANG=ASSEM,
PSBNAME=CSQQTRMN,  Runs program CSQQTRMN
CMPAT=YES
```

Figure 118. Example PSB definition for CSQQTRMN

For further information about starting and stopping the IMS trigger monitor, see Controlling the IMS trigger monitor.

z/OS Setting up the IMS bridge

The IBM MQ - IMS bridge is an optional component that enables IBM MQ to input and output to and from existing programs and transactions that are not IBM MQ-enabled.

This topic describes what you must do to customize the IBM MQ - IMS bridge.

Define the XCF and OTMA parameters for IBM MQ.
This step defines the XCF group and member names for your IBM MQ system, and other OTMA parameters. IBM MQ and IMS must belong to the same XCF group. Use the OTMACON keyword of the CSQ6SYSP macro to tailor these parameters in the system parameter load module.

See Using CSQ6SYSP for more information.

Define the XCF and OTMA parameters to IMS.
This step defines the XCF group and member names for the IMS system. IMS and IBM MQ must belong to the same XCF group.

Add the following parameters to your IMS parameter list, either in your JCL or in member DFSPBxxx in the IMS PROCLIB:

OTMA=Y
This starts OTMA automatically when IMS is started. (It is optional, if you specify OTMA=N you can also start OTMA by issuing the IMS command /START OTMA.)

GRNAME=
This parameter gives the XCF group name.

It is the same as the group name specified in the storage class definition (see the next step), and in the **Group** parameter of the OTMACON keyword of the CSQ6SYSP macro.

OTMANM=
This parameter gives the XCF member name of the IMS system.

This is the same as the member name specified in the storage class definition (see the next step).

Tell IBM MQ the XCF group and member name of the IMS system.
This is specified by the storage class of a queue. If you want to send messages across the IBM MQ - IMS bridge you must specify this when you define the storage class for the queue. In the storage class, you must define the XCF group and the member name of the target IMS system. To do this, either use the IBM MQ operations and control panels, or use the IBM MQ commands as described in Introduction to Programmable Command Formats.

Set up the security that you require.
The /SECURE OTMA IMS command determines the level of security to be applied to **every** IBM MQ queue manager that connects to IMS through OTMA. See Security considerations for using IBM MQ with IMS for more information.

Adding an additional IMS connection to the same queue manager

To add an IMS connection to the same queue manager you must define a second storage class (STGCLASS) to point at the new IMS; see DEFINE STGCLASS for more information.

Important:

- One local queue cannot point to two storage classes.
- One storage class cannot point to two IMS bridges.
- IBM MQ and IMS must belong to the same XCF group. Use the OTMACON keyword of the CSQ6SYSP macro to tailor these parameters in the system parameter load module.

  See Using CSQ6SYSP for more information.

Related concepts

IBM MQ and IMS

"Setting up the IMS adapter" on page 1005

To use IBM MQ within IMS requires the IBM MQ - IMS adapter (generally referred to as the IMS adapter).

IMS and IMS bridge applications on IBM MQ for z/OS

z/OS Using IBM MQ with CICS

To use IBM MQ with CICS, you must configure the IBM MQ CICS adapter and, optionally, the IBM MQ CICS bridge components.

For more information about configuring the IBM MQ CICS adapter and the IBM MQ CICS bridge components, see the Configuring connections to MQ section of the CICS documentation.

Related concepts

IBM MQ and CICS

"Using IBM MQ with IMS" on page 1005

The IBM MQ -IMS adapter, and the IBM MQ - IMS bridge are the two components which allow IBM MQ to interact with IMS.

Related reference

"Upgrading and applying service to Language Environment or z/OS Callable Services" on page 1013

The actions you must take vary according to whether you use CALLLIBS or LINK, and your version of SMP/E.

z/OS Upgrading and applying service to Language Environment or z/OS Callable Services

The actions you must take vary according to whether you use CALLLIBS or LINK, and your version of SMP/E.

The following tables show you what you need to do to IBM MQ for z/OS if you upgrade your level of, or apply service to, the following products:

- Language Environment
- z/OS Callable Services (APPC and RRS for example)

Table 69. Service has been applied or the product has been upgraded to a new release

Product	Action if using CALLLIBS and SMP/E V3r2 or later **Note: You do not need to run separate jobs for Language Environment and Callable services. One job will suffice.**	Action if using LINK
Language Environment	1. Set the Boundary on your SMP/E job to the Target zone. 2. On the SMPCNTL card specify LINK LMODS CALLLIBS. You can also specify other parameters such as CHECK, RETRY(YES) and RC. See *z/OS SMP/E Commands* for further information. 3. Run the SMP/E job.	No action required provided that the SMP/E zones were set up for automatic relinking, and the CSQ8SLDQ job has been run.
Callable Services	1. Set the Boundary on your SMP/E job to the Target zone. 2. On the SMPCNTL card specify LINK LMODS CALLLIBS. You can also specify other parameters such as CHECK, RETRY(YES) and RC. See *z/OS SMP/E Commands* for further information. 3. Run the SMP/E job.	No action required provided that the SMP/E zones were set up for automatic relinking, and the CSQ8SLDQ job has been run.

Table 70. One of the products has been updated to a new release in a new SMP/E environment and libraries

Product	Action if using CALLLIBS and SMP/E V3r2 or later **Note: You do not need to run three separate jobs for Language Environment and Callable services. One job will suffice for both products.**	Action if using LINK
Language Environment	1. Change the DDDEFs for SCEELKED and SCEESPC to point to the new library. 2. Set the Boundary on your SMP/E job to the Target zone. 3. On the SMPCNTL card specify LINK LMODS CALLLIBS. You can also specify other parameters such as CHECK, RETRY(YES) and RC. See *z/OS SMP/E Commands* for further information. 4. Run the SMP/E job.	1. Delete the XZMOD subentries for the following LMOD entries in the IBM MQ for z/OS target zone: CMQXDCST, CMQXRCTL, CMQXSUPR, CSQCBE00, CSQCBE30, CSQCBP00, CSQCBP10, CSQCBR00, CSQUCVX, CSQUDLQH, CSQVXPCB, CSQVXSPT, CSQXDCST, CSQXRCTL, CSQXSUPR, CSQXTCMI, CSQXTCP, CSQXTNSV, CSQ7DRPS, IMQB23IC, IMQB23IM, IMQB23IR, IMQS23IC, IMQS23IM, IMQS23IR 2. Set up the appropriate ZONEINDEXs between the IBM MQ zones and the Language Environment zones. 3. Tailor CSQ8SLDQ to refer to the new zone on the FROMZONE parameter of the LINK commands. CSQ8SLDQ can be found in the SCSQINST library. 4. Run CSQ8SLDQ.

Table 70. One of the products has been updated to a new release in a new SMP/E environment and libraries (continued)

Product	Action if using CALLLIBS and SMP/E V3r2 or later **Note: You do not need to run three separate jobs for Language Environment and Callable services. One job will suffice for both products.**	Action if using LINK
Callable services	1. Change the DDDEF for CSSLIB to point to the new library 2. Set the Boundary on your SMP/E job to the Target zone. 3. On the SMPCNTL card specify LINK LMODS CALLLIBS. You can also specify other parameters such as CHECK, RETRY(YES) and RC. See *z/OS SMP/E Commands* for further information. 4. Run the SMP/E job.	1. Delete the XZMOD subentries for the following LMOD entries in the IBM MQ for z/OS target zone: CMQXRCTL, CMQXSUPR, CSQBSRV, CSQILPLM, CSQXJST, CSQXRCTL, CSQXSUPR, CSQ3AMGP, CSQ3EPX, CSQ3REPL 2. Set up the appropriate ZONEINDEXs between the IBM MQ zones and the Callable Services zones. 3. Tailor CSQ8SLDQ to refer to the new zone on the FROMZONE parameter of the LINK commands. CSQ8SLDQ can be found in the SCSQINST library. 4. Run CSQ8SLDQ.

For an example of a job to relink modules when using CALLLIBS, see "Running a LINK CALLLIBS job" on page 1015.

z/OS Running a LINK CALLLIBS job

An example job to relink modules when using CALLLIBS.

The following is an example of the job to relink modules when using CALLLIBs on a SMP/E V3r2 system. You must provide a JOBCARD and the data set name of SMP/E CSI that contains IBM MQ for z/OS.

```
//*******************************************************************
//* RUN LINK CALLLIBS.
//*******************************************************************
//CALLLIBS EXEC PGM=GIMSMP,REGION=4096K
//SMPCSI  DD DSN=your.csi
//       DISP=SHR
//SYSPRINT DD SYSOUT=*
//SMPCNTL DD *
SET BDY(TZONE).
LINK LMODS CALLLIBS .
/*
```

Figure 119. Example SMP/E LINK CALLLIBS job

z/OS Using OTMA exits in IMS

Use this topic if you want to use IMS Open Transaction Manager Access exits with IBM MQ for z/OS.

If you want to send output from an IMS transaction to IBM MQ, and that transaction did not originate in IBM MQ, you need to code one or more IMS OTMA exits.

Similarly if you want to send output to a non-OTMA destination, and the transaction did originate in IBM MQ, you also need to code one or more IMS OTMA exits.

The following exits are available in IMS to enable you to customize processing between IMS and IBM MQ:

- An OTMA pre-routing exit
- A destination resolution user (DRU) exit

OTMA exit names

You must name the pre-routing exit DFSYPRX0. You can name the DRU exit anything, as long as it does not conflict with a module name already in IMS.

Specifying the destination resolution user exit name

You can use the *Druexit* parameter of the OTMACON keyword of the CSQ6SYSP macro to specify the name of the OTMA DRU exit to be run by IMS.

To simplify object identification, consider adopting a naming convention of DRU0xxxx, where xxxx is the name of your IBM MQ queue manager.

If you do not specify the name of a DRU exit in the OTMACON parameter, the default is DFSYDRU0. See DFSYDRU0 for more information.

Naming convention for IMS destination

You need a naming convention for the destination to which you send the output from your IMS program. This is the destination that is set in the CHNG call of your IMS application, or that is preset in the IMS PSB.

A sample scenario for an OTMA exit

Use the following topics for an example of a pre-routing exit and a destination routing exit for IMS:

- “The pre-routing exit DFSYPRX0” on page 1016
- “The destination resolution user exit” on page 1017

To simplify identification, make the OTMA destination name similar to the IBM MQ queue manager name, for example the IBM MQ queue manager name repeated. In this case, if the IBM MQ queue manager name is " **VCPE** ", the destination set by the CHNG call is " **VCPEVCPE** ".

Related concepts

IBM MQ and IMS

“Using IBM MQ with IMS” on page 1005

The IBM MQ -IMS adapter, and the IBM MQ - IMS bridge are the two components which allow IBM MQ to interact with IMS.

IMS and IMS bridge applications on IBM MQ for z/OS

z/OS The pre-routing exit DFSYPRX0

This topic contains a sample pre-routing exit for OTMA in IMS.

You must first code a pre-routing exit DFSYPRX0. See OTMA Destination Resolution user exit (DFSYPRX0 and other OTMAYPRX type exits) for parameters passed to this routine by IMS.

This exit tests whether the message is intended for a known OTMA destination (in our example VCPEVCPE). If it is, the exit must check whether the transaction sending the message originated in OTMA. If the message originated in OTMA, it will have an OTMA header, so you should exit from DFSYPRX0 with register 15 set to zero.

- If the transaction sending the message did not originate in OTMA, you must set the client name to be a valid OTMA client. This is the XCF member-name of the IBM MQ queue manager to which you want to send the message. You should set your client name (in the OTMACON parameter of the CSQ6SYSP

macro) is set to the queue manager name. This is the default. You should then exit from DFSYPRX0 setting register 15 to 4.

- If the transaction sending the message originated in OTMA, and the destination is non-OTMA, you should set register 15 to 8 and exit.
- In all other cases, you should set register 15 to zero.

If you set the OTMA client name to one that is not known to IMS, your application CHNG or ISRT call returns an A1 status code.

For an IMS system communicating with more than one IBM MQ queue manager, you should repeat the logic for each IBM MQ queue manager.

Sample assembler code is shown in Figure 120 on page 1017:

```
TITLE 'DFSYPRX0: OTMA PRE-ROUTING USER EXIT'
DFSYPRX0 CSECT
DFSYPRX0 AMODE 31
DFSYPRX0 RMODE ANY
*
SAVE (14,12),,DFSYPRX0&SYSDATE&SYSTIME
SPACE 2
LR  R12,R15        MODULE ADDRESSABILITY
USING DFSYPRX0,R12
*
L   R2,12(,R1)      R2 -> OTMA PREROUTE PARMS
*
LA  R3,48(,R2)      R3 AT ORIGINAL OTMA CLIENT (IF ANY)
CLC  0(16,R3),=XL16'00'  OTMA ORIG?
BNE  OTMAIN        YES, GO TO THAT CODE
*
NOOTMAIN DS 0H            NOT OTMA INPUT
LA  R5,8(,R2)       R5 IS AT THE DESTINATION NAME
CLC  0(8,R5),=C'VCPEVCPE' IS IT THE OTMA UNSOLICITED DEST?
BNE  EXIT0         NO, NORMAL PROCESSING
*
L   R4,80(,R2)      R4 AT ADDR OF OTMA CLIENT
MVC  0(16,R4),=CL16'VCPE' CLIENT OVERRIDE
B   EXIT4         AND EXIT
*
OTMAIN  DS 0H           OTMA INPUT
LA  R5,8(,R2)       R5 IS AT THE DESTINATION NAME
CLC  0(8,R5),=C'VCPEVCPE' IS IT THE OTMA UNSOLICITED DEST?
BNE  EXIT8         NO, NORMAL PROCESSING
```

```
*
EXIT0  DS 0H
LA  R15,0         RC = 0
B   BYEBYE
*
EXIT4  DS 0H
LA  R15,4         RC = 4
B   BYEBYE
*
EXIT8  DS 0H
LA  R15,8         RC = 8
B   BYEBYE
*
BYEBYE  DS 0H
RETURN (14,12),,RC=(15)   RETURN WITH RETURN CODE IN R15
SPACE 2
REQUATE
SPACE 2
END
```

Figure 120. OTMA pre-routing exit assembler sample

z/OS The destination resolution user exit

This topic contains a sample destination resolution user exit for IMS.

If you have set registers 15 to 4 in DFSYPRX0, or if the source of the transaction was OTMA ***and*** you set Register 15 to zero, your DRU exit is invoked. In this example, the DRU exit name is DRU0VCPE.

The DRU exit checks if the destination is VCPEVCPE. If it is, it sets the OTMA user data (in the OTMA prefix) as follows:

Offset
OTMA user data

(decimal)

0
OTMA user data length (in this example, 334)

2
MQMD

326
Reply to format

These offsets are where the IBM MQ - IMS bridge expects to find this information.

The DRU exit should be as simple as possible. Therefore, in this sample, all messages originating in IMS for a particular IBM MQ queue manager are put to the same IBM MQ queue.

If the message needs to be persistent, IMS must use a synchronized transaction pipe. To do this, the DRU exit must set the OUTPUT flag. See Specifying synchronized tpipes for IBM MQ for more information.

Write an IBM MQ application to process this queue, and use information from the MQMD structure, the MQIIH structure (if present), or the user data, to route each message to its destination.

A sample assembler DRU exit is shown in Figure 121 on page 1019.

```
TITLE 'DRU0VCPE: OTMA DESTINATION RESOLUTION USER EXIT'
DRU0VCPE CSECT
DRU0VCPE AMODE 31
DRU0VCPE RMODE ANY
*
SAVE (14,12),,DRU0VCPE&SYSDATE&SYSTIME
SPACE 2
LR  R12,R15            MODULE ADDRESSABILITY
USING DRU0VCPE,R12
*
L   R2,12(,R1)           R2 -> OTMA DRU PARMS
*
L   R5,88(,R2)           R5 ADDR OF OTMA USERDATA
LA  R6,2(,R5)           R6 ADDR OF MQMD
USING MQMD,R6            AS A BASE
*
LA  R4,MQMD_LENGTH+10       SET THE OTMA USERDATA LEN
STH  R4,0(,R5)           = LL + MQMD + 8
*                       CLEAR REST OF USERDATA
MVI  0(R6),X'00'          ...NULL FIRST BYTE
MVC  1(255,R6),0(R6)        ...AND PROPAGATE IT
MVC  256(MQMD_LENGTH-256+8,R6),255(R6) ...AND PROPAGATE IT
*
VCPE   DS  0H
CLC  44(16,R2),=CL16'VCPE'     IS DESTINATION VCPE?
BNE  EXIT4             NO, THEN DEST IS NON-OTMA
MVC  MQMD_REPLYTOQ,=CL48'IMS.BRIDGE.UNSOLICITED.QUEUE'
MVC  MQMD_REPLYTOQMGR,=CL48'VCPE'  SET QNAME AND QMGRNAME
MVC  MQMD_FORMAT,MQFMT_IMS     SET MQMD FORMAT NAME
MVC  MQMD_LENGTH(8,R6),MQFMT_IMS_VAR_STRING
*                        SET REPLYTO FORMAT NAME
B   EXIT0
*
EXIT0  DS  0H
LA  R15,0             SET RC TO OTMA PROCESS
B   BYEBYE             AND EXIT
*
EXIT4  DS  0H
LA  R15,4             SET RC TO NON-OTMA
B   BYEBYE             AND EXIT
*
BYEBYE  DS  0H
RETURN (14,12),,RC=(15)       RETURN CODE IN R15
SPACE 2
REQUATE
SPACE 2
CMQA  EQUONLY=NO
CMQMDA DSECT=YES
SPACE 2
END
```

Figure 121. Sample assembler DRU exit

z/OS Using IBM z/OSMF to automate IBM MQ

The IBM z/OS Management Facility (z/OSMF) provides system management functions in a task-oriented, web browser-based user interface with integrated user assistance, so that you can more easily manage the day-to-day operations and administration of your mainframe z/OS systems.

By streamlining some traditional tasks and automating others, z/OSMF can help to simplify some areas of z/OS system management.

Resources can be provisioned or de-provisioned, at a click of a button, from a user provided portal. z/OSMF provides REST APIs to help with this task.

The sample marketplace portal supplied with z/OSMF can also be used to provision and de-provision resources. Alternatively, more experienced users can use the z/OSMF Web User Interface (WUI).

This section assumes that you understand z/OSMF, but if you are unfamiliar with z/OSMF you should read Getting started with z/OSMF. Alternatively, you can access this section from the z/OSMF WUI online help.

You should familiarize yourself with z/OS Cloud configuration, that is:

- Cloud Provisioning - Resource management services
- Workload Management - see IBM z/OS Management Facility Programming Guide for more information.
- Getting started - see *Getting Started Tutorial - Cloud*

z/OSMF 2.2 introduces role based activities and tasks, so it is important that you understand concepts like:

domains
administrators
approvers
tenants
templates
instances
workflows

and so on.

Sample IBM MQ z/OSMF workflows and associated files are provided, and can be installed as part of the IBM MQ for z/OS UNIX System Services Components feature. The installation process for this feature, and the directory and file structure, are described in the IBM MQ for z/OS Program Directory. Para enlaces de descarga de los directorios de programas, consulteIBM MQ for z/OS Archivos PDF del directorio de programas .

The sample workflows are written in XML and demonstrate how to automate the provisioning (creation) or de-provisioning (destruction) of IBM MQ queue managers, channel initiators, and local queues, and how to perform actions against the provisioned IBM MQ resources. Steps within the workflows submit jobs (JCL), run REXX execs, process Shell scripts, or issue REST API calls.

The samples are designed to illustrate the types of function that can be achieved using z/OSMF. It is anticipated that z/OSMF workflows will generally be used to provision resources and actions like put or get message will, in essence, be performed using IBM MQ applications.

You can run the sample workflows as supplied, provided the workflow variable properties have been set (as discussed in the following sections), or you can customize them as required. You might prefer to write your own workflows to perform additional function. Before running the sample workflows see:

- “Prerequisites for z/OSMF” on page 1020
- “Security settings ” on page 1022
- “Limitations ” on page 1024

Sample workflow applications are provided to:

- “Automate the provisioning or de-provisioning of IBM MQ queue managers and perform actions against the provisioned queue managers” on page 1025
- “Automate the provisioning or de-provisioning of IBM MQ local queues and perform actions against the provisioned queues” on page 1026.

Related concepts

“Setting up IBM MQ for z/OS” on page 901
Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

z/OS Prerequisites for z/OSMF

The prerequisites you require to run IBM z/OS Management Facility (z/OSMF) with IBM MQ

The workflows shipped in IBM MQ for z/OS 9.1.0 exploit new function in z/OSMF, which is provided through APARs on both z/OS 2.1 and 2.2. More details are provided in the following text.

1. You have installed and configured IBM z/OS Management Facility 2.2 correctly. If you are running with security enabled, ensure that all security settings as documented by z/OSMF have been configured.
2. You have installed the following APARs for:

   z/OS 2.1

   - PI71068
   - PI71079
   - PI71082
   - PI71084
   - OA50130

   z/OS 2.2

   - PI70526
   - PI70521
   - PI70527
   - PI67839
   - PI70767
   - PI46315
   - OA49081
   - OA49802
   - OA50130
3. The z/OSMF angel (if required) and server processes have been configured.
4. The z/OS Cloud environment has been configured (as briefly discussed above and documented by z/OSMF)
5. IBM MQ for z/OS 9.0.1 has been installed and the product load libraries are available.
6. The following IBM MQ queue manager customization tasks have been performed:

Task	Description
1	Identify the z/OS system parameters
2	APF authorize the IBM MQ load libraries
3	Update the z/OS link list and LPA
4	Update the z/OS program properties table

7. The sample workflows and associated files are installed in a suitable z/OS UNIX System Services (z/OS UNIX) directory.
8. The `/tmp` z/OS UNIX directory is available, because the provision.xml workflow might create a temporary file in this directory. If a file is created, the workflow, in general, deletes the file after use.
9. The `deprovision.xml` file has steps in it that invoke the CSQ4ZWS1.rexx and CSQ4ZWS2.rexx REXX execs. These execs wait for the queue manager and channel initiator subsystems to stop; the execs invoke the z/OS UNIX **SLEEP** command as a system call.

   Depending on your z/OS UNIX configuration, you might find that the **SLEEP** command does not work as coded. If, during processing you encounter an error which indicates that the **SLEEP** command cannot be found, you can try replacing the following lines in execs CSQ4ZWS1.rexx and CSQ4ZWS2.rexx:

```
CALL SYSCALLS('ON')            /* Enable z/OS UNIX calls    */
ADDRESS SYSCALL
```

```
"SLEEP" 10                              /* Sleep for 10 seconds */
CALL SYSCALLS 'OFF'                  /* Disable z/OS UNIX calls    */
```

with

```
'sleep' 10
```

Then, issue the Open MVS (OMVS) **env** command to check your PATH environment variable setting. Ensure that the directory which contains the **sleep** command is defined to the PATH. Note that the **sleep** command is typically found in the `/bin` directory.

10. Ensure that z/OSMF has been started.

    Both the angel and server z/OSMF processes must be started and the z/OSMF Web User Interface (WUI) be up and running. For further details, see Liberty profile: Process types on z/OS.

    Even if you intend to drive the workflows using the REST API, the z/OSMF WUI needs to be started. The z/OSMF WUI can be useful for monitoring the creation and execution of workflows.

Related concepts

"Using IBM z/OSMF to automate IBM MQ " on page 1019
The IBM z/OS Management Facility (z/OSMF) provides system management functions in a task-oriented, web browser-based user interface with integrated user assistance, so that you can more easily manage the day-to-day operations and administration of your mainframe z/OS systems.

z/OS Security settings

The security settings required to run z/OSMF.

The following User ID variable properties are defined in the properties file. For more details, see "Running the workflows" on page 1029.

User ID property	Description
CSQ_USERID	User ID used to run the workflow steps. Note, however, that selected steps (which generally require an elevated level of authority) will be run with different user IDs based on the setting of the **CSQ_ADMIN_*** user IDs listed in the following text. The user ID in use is identified by the **runAsUser** property on the respective step in the workflows.
CSQ_ADMIN_APF_USERID	User ID to use when APF authorizing the load library that contains the queue manager system parameter module.
CSQ_APF_APPROVAL_ID	The approval ID used to permit users to run the data set APF authorization step as user CSQ_ADMIN_APF_USERID.
CSQ_ADMIN_CONSOLE_USERID	User ID used when running steps under the run that issue z/OS console commands. **Attention:** This user ID needs to be permitted UPDATE access to the started task profile (MVS.START.STC.*) in the OPERCMDS class. See Controlling the use of operator commands in the z/OS documentation for more information.
CSQ_CONSOLE_APPROVAL_ID	The approval ID used to permit users to run steps that issue z/OS console commands under the run as user CSQ_ADMIN_CONSOLE_USERID.
CSQ_ADMIN_SAF_USERID	User ID to use when issuing SAF commands.
CSQ_SAF_APPROVAL_ID	The approval ID used to permit users to run the SAF command steps under the run as user CSQ_ADMIN_SAF_USERID.
CSQ_ADMIN_SSI_USERID	User ID to use when issuing the SETSSI command to identify the subsystem being provisioned to z/OS.

User ID property	Description
CSQ_SSI_APPROVAL_ID	The approval ID used to permit users to run the SETSSI command step under the run as user CSQ_ADMIN_SSI_USERID.

Note: The User ID being used to run the provision and de-provision workflows needs to have sufficient authority as listed below:

1. The Queue Manager provision and de-provision workflows use the SETPROG command to APF authorize data sets. Either the user ID is set in property CSQ_ADMIN_APF_USERID, or the user ID being used to run the workflows needs to be permitted to issue this command. You can achieve this by issuing the following command:

```
PERMIT MVS.SETPROG CLASS(OPERCMDS) ID(value of CSQ_ADMIN_APF_USERID) ACCESS(UPDATE)
```

   Note: The SETPROG command might not persist across an IPL of a z/OS system so, it might be necessary to manually issue the following SETPROG command following an IPL:

```
SETPROG APF,ADD,DSN=value of CSQ_AUTH_LIB_HLQ.value of CSQ_SSID.APF.LOAD,SMS
```

   For more details about the SETPROG command, see Using RACF to control APF lists.

   In addition, you might have enabled FACILITY class to control which libraries can be APF authorized, so you might need to issue the command:

```
PERMIT CSVAPF.libname CLASS(FACILITY) ID(value of CSQ_ADMIN_APF_USERID)
       ACCESS(UPDATE)
```

2. A step in the Queue Manager provision workflow issues the SETSSI command to identify the IBM MQ subsystem to z/OS. The User ID set in property CSQ_ADMIN_SSI_USERID needs to be permitted to use this command. You can achieve this by issuing the following command:

```
PERMIT MVS.SETSSI.ADD CLASS(OPERCMDS) ID(value of CSQ_ADMIN_SSI_USERID)
        ACCESS(CONTROL)
```

   Note: Subsystems that have been identified to z/OS through the SETSSI command do not persist across an IPL of a z/OS system. So, it might be necessary to manually issue the following SETSSI command following an IPL:

```
SETSSI  ADD,S='value of CSQ_SSID',I=CSQ3INI,
       P='CSQ3EPX,value of CSQ_CMD_PFX,S'
```

   For more details about the SETSSI command, see: SETSSI command.

3. The workflows issue queue manager commands, so if you are planning to enable security, the user ID set in property CSQ_ADMIN_RACF_USERID (or the user ID being used to run the workflows) needs to be granted CLAUTH (client authentication) authority to the MQADMIN or the MXADMIN class (depending on which class is being used). This is to allow this user ID to define security profiles to these classes. You can achieve this by issuing the following command:

```
ALTUSR value of CSQ_ADMIN_RACF_USERID CLAUTH(MQADMIN)
```

   For more details about **CLAUTH** see The CLAUTH (class authority) attribute.

4. The deprovision.xml workflow issues z/OS commands, for example, DISPLAY ACTIVE jobs, CANCEL or FORCE subsystems, so the user ID set in property CSQ_ADMIN_CONSOLE_USERID (or the user ID being used to run the workflows) needs to have suitable authority to issue such commands.
5. Users requesting a queue manager instance, using the templates table of the Software Services task, must have permission to access z/OSMF and the Configuration Assistant, as defined by z/OSMF.

6. The user ID of the consumer provisioning a queue manager requires authority to add and delete members from the PROCLIB data set defined with variable CSQ_PROC_LIB.
7. A queue manager must be provisioned ahead of provisioning queues.
8. To use the `queueLoad.xml` and `queueOffload.xml` workflows, the data sets used need to be defined ahead of time. Also, the user ID used to run these workflows needs to be granted UPDATE authority to the data sets.
9. A step in the queue manager `provision.xml` workflow currently disables subsystem security. You can modify Job `csq4znse.jcl` to enable subsystem security by adding the appropriate security commands for protecting IBM MQ resources. However, note that if you do add additional commands, you also need to add commands to delete security permissions in `csq4dse.jcl`, which is submitted by the deprovision.xml workflow.

   Note: This step issues RACF security commands. If you are using an alternate security product, you need to modify this step to issue the appropriate commands for your security product.

Network Requirements

When adding a queue manager template, and resources for the template, you need to click **Create network resource pool**. This creates a resource pool with network resources for this template.

Using the Configuration Assistant, your network administrator needs to complete this network resource pool definition by defining a limit for the number of ports that are to be allocated for this template.

For each template instance, the `provision.xml` workflow allocates a port in the range, and starts a listener to listen on that port.

Classifying with IBM Workload Manager

If you want to classify the queue manager and channel initiator address spaces with WLM, you need to specify this when adding a template for provisioning a queue manager.

Whether to classify or not, is controlled by flags **CSQ_DEFINE_MSTR_WLM_RULE** and **CSQ_DEFINE_CHIN_WLM_RULE**, which are set in file `workflow_variables.properties`.

For more information about classifying with WLM, refer to the *z/OSMF Configuration Guide*.

Related concepts

“Prerequisites for z/OSMF” on page 1020
The prerequisites you require to run IBM z/OS Management Facility (z/OSMF) with IBM MQ

z/OS Limitations

Limitations when using z/OSMF with IBM MQ.

1. The `provision.xml` workflow currently automates the following highlighted queue manager customization tasks:

Task	Description
1	Identify the z/OS system parameters
2	APF authorize the IBM MQ load libraries **(provision.xml does APF authorize some libraries)**
3	Update the z/OS link list and LPA
4	Update the z/OS program properties table
5	**Define the IBM MQ subsystem to z/OS**
6	**Create procedures for the IBM MQ queue manager**
7	**Create procedures for the channel initiator**

Task	Description
8	**Define the IBM MQ subsystem to a z/OS WLM service class**
9	Select and set up your coupling facility offload storage environment
10	Set up the coupling facility
11	Implement your ESM security controls
12	Update SYS1.PARMLIB members
13	**Customize the initialization input data sets**
14	**Create the bootstrap and log data sets**
15	**Define your page sets**
16	Add the IBM MQ entries to the Db2 data-sharing group
17	**Tailor your system parameter modules (some)**
18	**Tailor the channel initiator parameters (some)**
19	Set up Batch, TSO, and RRS adapters
20	Set up the operations and control panels
21	Include the IBM MQ dump formatting member
22	Suppress information messages
23	Update your system DIAG member for Advanced Message Security
24	Create procedures for Advanced Message Security
25	Set up the started task user Advanced Message Security
26	Grant RACDCERT permissions to the security administrator for Advanced Message Security
27	Grant users resource permissions for Advanced Message Security

2. Customization tasks that are not highlighted in bold text need to be performed manually, if required.
3. The sample INP1 and INP2 members are currently used as is. If required, additional properties can be defined to control the resources defined by these members.
4. Comments pertaining to specific properties listed in the properties file indicate any limitations of using those properties. For more details, see “Running the workflows” on page 1029.

Related concepts

“Security settings ” on page 1022
The security settings required to run z/OSMF.

z/OS Automate the provisioning of IBM MQ objects

Samples are supplied to automate the provisioning of queue managers and local queues.

Automate the provisioning or de-provisioning of IBM MQ queue managers and perform actions against the provisioned queue managers

The following queue manager specific sample z/OSMF workflows are provided:

Workflow name	Description
provision.xml	Provision an IBM MQ for z/OS queue manager

Workflow name	Description
	This sample workflow: • Provisions the required system resources for a queue manager. • Provisions the required system resources for a channel initiator. • Starts the queue manager (which also starts the channel initiator and TCP/IP listener) • Runs the sample queue manager installation verification program. An environment property can be set to control the provisioning of queue managers with different characteristics. For more information, see "Running the workflows" on page 1029. **Note:** A manifest file (`provision.mf`) is provided to assist with adding a template for this workflow. This file contains a reference to the **qaas_readme.pdf** file which contains additional information. You can access the file through a link, once the template has been added.
deprovision.xml	De-provision an IBM MQ for z/OS queue manager This sample workflow: • Stops the channel initiator (which also stops the TCP/IP listener) and the queue manager. • Waits for the subsystems to stop • De-provisions all channel initiator and queue manager system resources.
startQMgr.xml	Start an IBM MQ for z/OS queue manager This sample workflow starts the queue manager (which also starts the channel initiator and TCP/IP listener).
stopQMgr.xml	Stop an IBM MQ for z/OS queue manager This sample workflow stops the channel initiator (which also stops the TCP/IP listener) and the queue manager.

Each workflow performs one or more steps. Comments in the workflows explain the function performed by each step. Some of the steps just request data input, while some steps submit JCL, invoke REXX execs, Shell scripts, or issue REST API calls to accomplish the stated function.

Refer to each step for the exact name of the JCL or REXX exec files. The workflows and associated JCL or REXX exec files reference variables that are declared in one or more variable XML files. For more details, see "Workflow variable declaration files" on page 1028.

deprovision, **startQMgr**, and **stopQMgr** can be performed as actions against a provisioned IBM MQ for z/OS queue manager.

Automate the provisioning or de-provisioning of IBM MQ local queues and perform actions against the provisioned queues

The following queue specific sample z/OSMF workflows are provided:

Workflow name	Description
defineQueue.xml	Define a local queue This sample workflow demonstrates how z/OSMF workflows can be used to define small, medium, or large sized queues based on property settings.

Workflow name	Description
	Note: A manifest file (`provision.mf`) is provided to assist with adding a template for this workflow. This file contains a reference to the **qaas_readme.pdf** file which contains additional information. You can access the file through a link, once the template has been added.
displayQueue.xml	Display selected attributes of a local queue This sample workflow displays selected attributes of a local queue. The attributes are returned in a z/OSMF variable (refer to the steps in the workflow for the name of the variable) and subsequently displayed. If required, the contents of the variable can be accessed using a REST API. For more details, refer to Cloud provisioning REST APIs, and also see z/OSMF workflow services.
deleteQueue.xml	Delete a local queue This sample workflow deletes a local queue on a specified queue manager.
putQueue.xml	Put one or more messages to a local queue. This sample workflow puts one or more messages to a local queue. The message text can be specified but if more than one message is put to a local queue at the same time, the same message text is used.
getQueue.xml	Get one or more messages from a local queue. This sample workflow gets one or more messages from a local queue. The messages are returned in a z/OSMF variable (refer to the steps in the workflow for the name of the variable) and subsequently displayed. If required, you can access the contents of the variable using a REST API. For more details, refer to Cloud provisioning REST APIs, and also see z/OSMF workflow services.
loadQueue.xml	Load messages from a data set to a local queue. This sample workflow loads messages from a data set on to a local queue. The default name of the data set is specified by setting a property. For more details, see “Running the workflows” on page 1029.
offloadQueue.xml	Offload messages from a local queue to a data set. This sample workflow off-loads messages from a local queue to a data set. The default name of the data set is specified by setting a property. For more details, see “Running the workflows” on page 1029.
clearQueue.xml	Clear messages on a local queue. This sample workflow clears (deletes) all messages on a local queue.

Notes:

1. The **Put Queue** action allows you to enter some message data and put one or more messages onto a queue. If more than one message is to be placed onto a queue during a given request, the same message data is used.
2. The loadQueue.xml and offloadQueue.xml workflows invoke the executable module, CSQUDMSG in the SCSQLOAD library, with an alias of QLOAD. This is equivalent to the **dmpmqmsg** utility available with IBM MQ for Multiplatforms. Therefore messages loaded from a data set onto a queue, or from a queue onto a data set, are expected to be in the **dmpmqmsg** format.

Sample JCL is also provided as member CSQ4QLOD in SCSQPROC.

The easiest way to try out the loadQueue and offloadQueue actions is to do the following:

a. Issue **putQueue** a few times to put some messages on to a queue.

b. Use **offloadQueue** to offload the messages from the queue on to a data set.

c. If required, issue **clearQueue** to remove all messages from the queue.

d. Use **loadQueue** to load the messages from a data set onto the same or a different queue.

If you are interested in the **dmpmqmsg** format, you can browse the contents of the data set, once you have issued an Offload request.

3. You can perform **displayQueue**, **deleteQueue**, **putQueue**, **getQueue**, **loadQueue**, **offloadQueue**, and **clearQueue** as actions against a provisioned IBM MQ for z/OS local queue. For further details about actions and action files, refer to the *z/OSMF Programming Guide*.
4. All action related workflows are deleted by default. The reason for this is to minimize the need for users to cleanup workflows.

   The problem with this however is that where an action results in some output. For example, the **displayQueue** and **getQueue** actions both produce output.

   The output cannot be seen since the related workflow is deleted as soon as the action has been performed. So, if you drive the workflow actions from the z/OS WUI, you need to set the **cleanAfterComplete** flag to *false* on the **<workflow>** tag for each action whose output you want to see.

   For example, to see the output of **displayQueue**, set the flag as follows:

```
<action name="displayQueue">
  <workflow cleanAfterComplete="false">
  ...

  ...
  </workflow>
</action>
```

   However, this means that you then have to manually clean up action related workflows.

Each sample z/OSMF workflow performs one or more steps. Comments in the workflows explain the function performed by each step. Some of the steps just request data input while some steps submit JCL and others invoke REXX execs to accomplish the stated function.

Refer to each step for the exact name of the JCL or REXX exec files. The workflows and associated JCL or REXX exec files reference variables that are declared in one or more “Workflow variable declaration files” on page 1028.

Related concepts

“Limitations ” on page 1024
Limitations when using z/OSMF with IBM MQ.

z/OS Running workflows

A description of the files referenced by the sample The z/OSMF workflows, and how you run a workflow.

Workflow variable declaration files

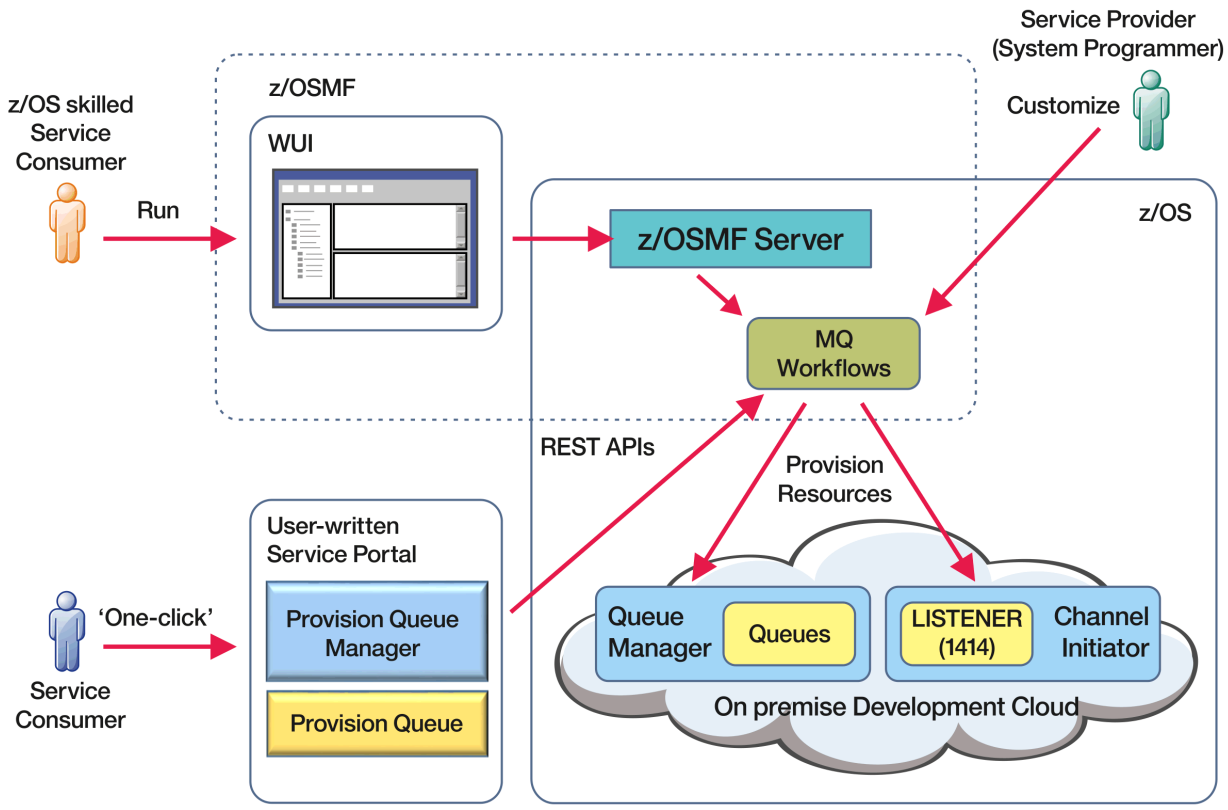
The following files declare variables that are referenced by the sample z/OSMF workflows and associated JCL or REXX exec files:

Workflow variable declaration file name	Description
common_variables.xml	Variables common to both the queue manager (plus channel initiator) and queue workflows.
qmgr_variables.xml	Variables specific to the queue manager (plus channel initiator) workflows.
queue_variables.xml	Variables specific to the queue workflows.
tcpip_variables.xml	Variables specific to the queue manager (plus channel initiator) workflows, and used for identifying TCP/IP resources.

Note: The default visibility of variables is *private*. To allow variables to be queried using the z/OSMF REST API, selected variables have been marked as *public*. However, you can change the visibility of a given variable if required.

Running the workflows

Figure 122. 'One-click' provisioning of IBM MQ for z/OS resources



Before the workflows can be run, some properties need to be set in the following file:

Workflow variable properties file name	Description
workflow_variables.properties	Initial properties for the workflow variables. Comments in the file indicate the purpose of each property. • Properties within meta-brackets (< >) need to be set to user specific values.

Workflow variable properties file name	Description
	• An environment property can be set to provision queue managers for development (DEV), or test (TEST), or quality assurance (QA), or production (PROD) environments. Additional property settings control the characteristics of the queue manager to be provisioned for each environment. For example you can vary the number of active logs, or the number of page sets, for each environment type. • Other properties are set to IBM MQ default values but can be modified to meet local conventions if required.

In general, once the properties have been set, the workflows can be run as is. However, if required, you can customize a workflow to modify or remove existing steps, or to add new steps.

Workflows can be run:

- From the z/OSMF WUI.

  From Cloud Provisioning -> Software Services in the WUI, workflows can be run in automatic or manual mode. The manual mode is useful when testing, and in both modes the progress of each step in the workflow can be monitored.

  For more details, see Cloud provisioning services and Create a workflow.
- Using the z/OSMF REST Workflow Services.

  The REST Workflow Services can be used to run workflows through a REST API. This mode is useful for creating one-click operations from a user-written portal.

  For more details, refer to Cloud provisioning REST APIs, and also see z/OSMF workflow services.
- Using the sample marketplace portal provided with z/OSMF.

Related concepts

"Automate the provisioning of IBM MQ objects" on page 1025
Samples are supplied to automate the provisioning of queue managers and local queues.

z/OS MQ Adv. VUE Habilitación de agentes de MFT para conectarse a gestores de colas remotos de z/OS

Sujeto a titularidad, un agente de MFT en z/OS puede utilizar una conexión de cliente para conectarse a un gestor de colas de z/OS . Esto puede dar lugar a topologías de IBM MQ más sencillas.

Si un agente de MFT en z/OS está asociado con el identificador de producto (PID) de IBM MQ Advanced for z/OS VUE o IBM MQ Advanced for z/OS, el agente puede utilizar una conexión de cliente para conectarse a un gestor de colas en z/OS.

Para obtener información sobre los distintos PID, consulte Identificadores de producto deIBM MQ e información de exportación. Para obtener información sobre cómo establecer el PID asociado a una instalación de MFT , consulte **`fteSetProductId`**.

El PID bajo el cual se está ejecutando el agente se muestra en el registro durante el inicio del agente.

Un agente de MFT en z/OS, que se ejecuta bajo cualquier otro PID, sólo puede conectarse a un gestor de colas local utilizando una conexión en modalidad de enlaces.

Si un agente intenta conectarse a un gestor de colas que no se está ejecutando en z/OS, se emite el mensaje BFGQM1044E y finaliza el inicio del agente.

Tareas relacionadas

Inicio de un agente MFT en z/OS

Detención de un agente MFT en z/OS

Referencia relacionada

El archivo MFT `agent.properties`

Configuración de IBM MQ Internet Pass-Thru

En esta sección se describen las distintas características que admite IBM MQ Internet Pass-Thru (MQIPT) y cómo configurarlas.

Configure MQIPT realizando cambios en el archivo de configuración `mqipt.conf`. La estructura del archivo de configuración de MQIPT y las propiedades que se pueden especificar se describen en Referencia de configuración de IBM MQ Internet Pass-Thru.

Nota: Debe establecer permisos de archivo seguro en el directorio en el que se encuentra el archivo `mqipt.conf` para evitar que los usuarios no autorizados vean las contraseñas almacenadas o cambien la configuración. Proteja todas las contraseñas especificadas en el archivo de configuración siguiendo el procedimiento de “Cifrado de contraseñas almacenadas en MQIPT” en la página 1071.

Los cambios en el archivo de configuración entran en vigor cuando MQIPT se inicia o se renueva. El hecho de renovar una instancia activa de MQIPT hace que los cambios de configuración entren en vigor sin reiniciar MQIPT. Cuando MQIPT se renueva, el archivo de configuración de `mqipt.conf` se vuelve a leer y MQIPT realiza las acciones siguientes:

- Las rutas activas que están marcadas como inactivas o que ya no se especifican en el archivo de configuración, se cierran y ya no aceptan conexiones entrantes.
- Las rutas que están marcadas como activas en el archivo de configuración, y no se están ejecutando actualmente, se inician.
- Se aplican los cambios en los parámetros de configuración de las rutas activas. Cuando sea posible, estos cambios entran en vigor sin ninguna interrupción en las conexiones activas. Para algunos cambios de parámetro, tales como un cambio en el destino de ruta, se cierran todas las conexiones antes de aplicar el cambio y se reinicia la ruta.

Para renovar MQIPT, utilice el mandato **`mqiptAdmin`**. Para obtener más información sobre cómo administrar MQIPT utilizando el mandato **`mqiptAdmin`**, consulte Administración de MQIPT utilizando la línea de mandatos.

Soporte HTTP en MQIPT

MQIPT da soporte al tunelado HTTP. MQIPT se puede configurar de modo que los paquetes de datos que reenvía se codifican como solicitudes HTTP.

Los canales de IBM MQ no aceptan solicitudes HTTP. Por consiguiente, se necesita un segundo MQIPT para recibir las solicitudes HTTP y convertirlas de nuevo en paquetes de protocolo IBM MQ. El segundo MQIPT elimina la cabecera HTTP para convertir el paquete entrante de nuevo en un paquete de protocolo IBM MQ estándar, antes de pasarlo al gestor de colas de destino.

Nota: V 9.4.0 V 9.4.0 A partir de IBM MQ 9.4.0, las rutas de MQIPT no aceptan conexiones HTTP de forma predeterminada. Las rutas deben estar configuradas para aceptar conexiones HTTP utilizando la propiedad **`AllowedProtocols`** .

Cuando HTTP se está utilizando entre dos instancias de MQIPT, la conexión TCP/IP en la cual fluyen solicitudes y respuestas HTTP es persistente y se mantiene abierto durante el ciclo de vida del canal de mensajes. MQIPT no cierra la conexión TCP/IP entre los pares de solicitud/respuesta.

Si dos instancias de MQIPT se comunican a través HTTP, es posible que una solicitud HTTP permanezca pendiente durante un periodo prolongado. Un ejemplo es un canal de solicitante/servidor, cuando el lado del servidor está a la espera de que lleguen nuevos mensajes en su cola de transmisión. El protocolo de canal IBM MQ proporciona unun mecanismo de "latidos", que requiere que se finalice la espera de forma periódica para enviar mensajes de pulsaciones a su socio. El periodo de latido de canal predeterminado es 5 minutos. MQIPT utiliza este latido como respuesta HTTP. No inhabilite este latido de canal, o

establézcalo en un valor excesivamente alto, para evitar causar problemas con tiempos de espera en algunos cortafuegos.

MQIPT acepta el tráfico HTTP en formato fragmentado, generado por un servidor o proxy HTTP.

Para ver un ejemplo de cómo utilizar HTTP en MQIPT, consulte Configuración de túneles HTTP.

Proxies HTTP

Un proxy HTTP se puede colocar entre las dos instancias de MQIPT. El proxy HTTP debe cumplir los requisitos siguientes:

- El proxy debe dar soporte al protocolo HTTP 1.1.
- El proxy debe respetar las cabeceras HTTP **Connection** o **Proxy-Connection** establecidas por MQIPT . Esto permite que las conexiones entre las dos instancias de MQIPT se mantengan abiertas durante el tiempo de vida del canal de mensajes.
- Se debe mantener una correlación de uno a uno de conexiones persistentes en todo el proxy. Esto asegura que las conexiones TCP/IP del proxy al MQIPT de destino no se utilicen para transmitir datos para más de un canal de mensajes.

Puede establecer propiedades para configurar cómo se gestionan las conexiones persistentes en algunos proxies HTTP. Por ejemplo, es posible que pueda establecer el número máximo de solicitudes que se pueden realizar en una conexión persistente. Se deben establecer las propiedades siguientes:

- Las conexiones persistentes deben habilitarse.
- La reutilización de conexiones TCP/IP desde el proxy a MQIPT por más de una sesión HTTP debe inhabilitarse, para mantener una correlación de uno a uno de conexiones persistentes a través del proxy.
- El tiempo de espera en las solicitudes de proxy debe establecerse en un valor alto. Por ejemplo, 12 horas.
- El número máximo de solicitudes que se pueden realizar en una conexión persistente debe establecerse en un valor alto. por ejemplo, 5000.

MQIPT utiliza solicitudes HTTP POST para enviar datos entre las dos instancias de MQIPT. Si la configuración de MQIPT especifica el nombre de host del proxy utilizando la propiedad **HTTPProxy**, MQIPT se conecta al proxy y utiliza el método HTTP CONNECT para solicitar que el proxy establezca un túnel en el MQIPT de destino. Esto permite que las conexiones HTTPS pasen a través del proxy sin terminar la sesión TLS en el proxy.

Si se coloca un equilibrador de carga entre las instancias de MQIPT, debe configurarse para utilizar el valor de la cookie HTTP *MQIPTSessionId* para asegurarse de que todas las solicitudes de cada sesión se reenvían al mismo destino.

HTTPS en MQIPT

HTTPS se puede utilizar en una conexión HTTP habilitando las propiedades de ruta **HTTPS** y **SSLClient** en MQIPT que emite la conexión de cliente.

MQIPT debe tener acceso al certificado CA de confianza que se utilizará para autenticarse con el proxy/servidor HTTP de destino. La propiedad **SSLClientCAKeyring** se puede utilizar para definir el archivo de conjunto de claves que contiene el certificado CA de confianza.

Una configuración común para HTTPS utilizará un proxy HTTP local para crear túneles a través de un cortafuegos y conectarse a un servidor HTTP remoto (u otro proxy) que, a su vez, se conectará al MQIPT remoto. Este MQIPT en el lado del servidor de la conexión no necesita ninguna configuración específica, ya que la solicitud de conexión se trata como cualquier conexión HTTP normal.

MQIPT utiliza las propiedades **HTTPProxy** y **HTTPServer** para distinguir los proxies locales y remotos. La ruta de MQIPT con la propiedad **HTTPProxy** establecida se ve como el proxy HTTP local y la ruta de MQIPT con la propiedad **HTTPServer** establecida es el servidor remoto (o proxy).

Las conexiones HTTPS se realizan normalmente con la dirección de puerto de escucha 443 en el servidor/proxy/HTTP, pero las propiedades **HTTPProxyPort** y **HTTPServerPort** se pueden utilizar para alterar temporalmente este valor predeterminado.

Soporte de SOCKS en MQIPT

Un proxy SOCKS es un servicio de red utilizado como punto de salida controlado a través de un cortafuegos. Una aplicación con SOCKS habilitado, que se ejecuta dentro del cortafuegos, puede utilizar el proxy SOCKS para conectarse a una aplicación remota.

MQIPT puede actuar como un proxy SOCKS habilitando la propiedad **SocksServer**, permitiendo así que una aplicación IBM MQ habilitada para SOCKS se conecte a través de MQIPT a un gestor de colas IBM MQ remoto. Cuando se utiliza esta característica, el destino y la dirección de puerto de destino se obtienen durante el proceso de reconocimiento SOCKS y, por lo tanto, se alteran temporalmente las propiedades de ruta **Destination** y **DestinationPort**. Esta es una característica clave para dar soporte a la agrupación en clúster de IBM MQ.

MQIPT también puede actuar como un cliente SOCKS, en nombre de una aplicación IBM MQ local que no haya habilitado SOCKS. Esto es útil cuando se utiliza un cortafuegos que permite conexiones de salida solo a través de un proxy SOCKS. Cada ruta de MQIPT se puede configurar para comunicarse con un proxy SOCKS diferente.

Consulte Configuración de un proxy SOCKS para ver un ejemplo de cómo utilizar SOCKS.

Agrupación en clúster en MQIPT

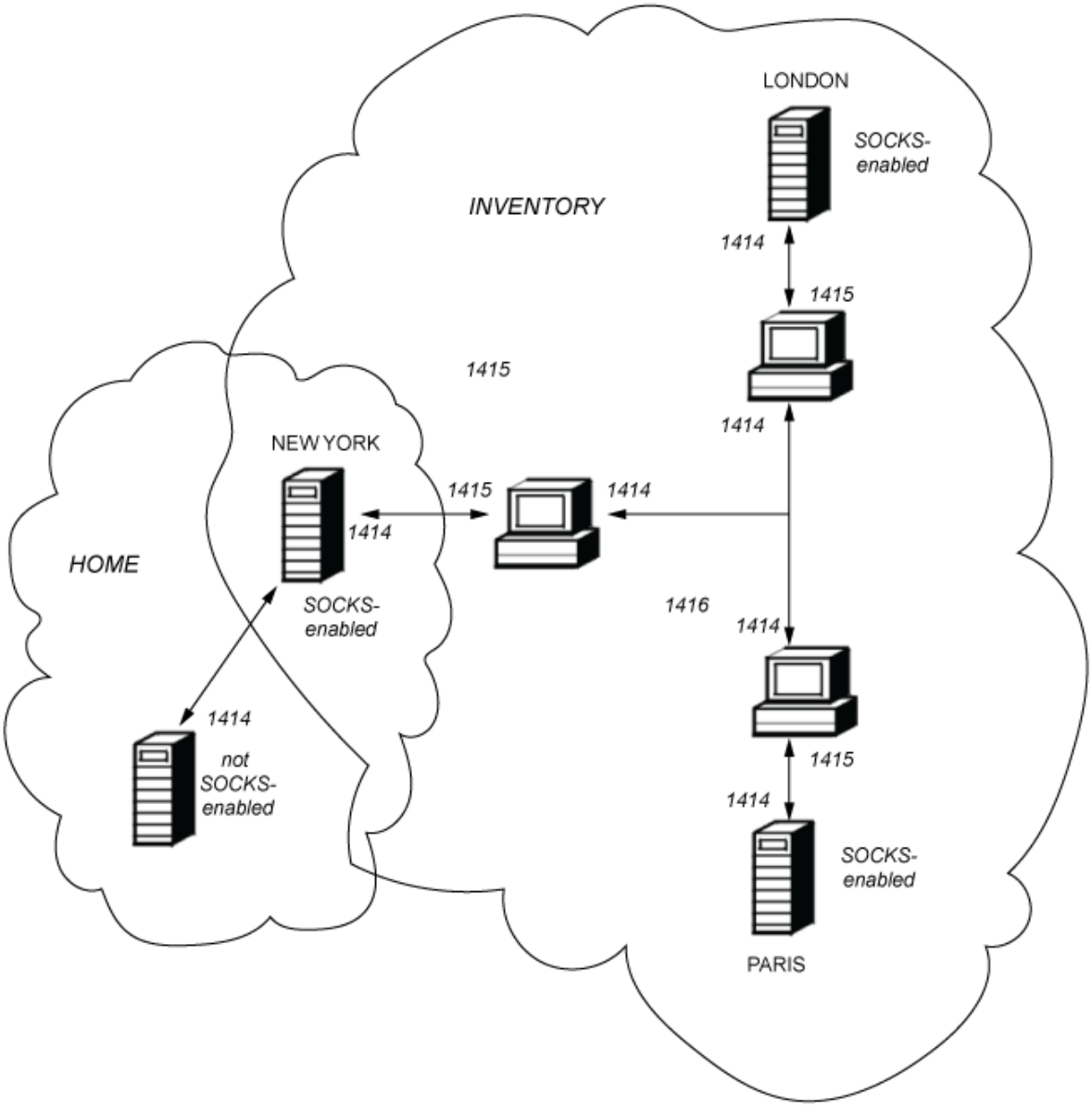
Los clústeres de IBM MQ se pueden utilizar con MQIPT mediante la habilitación de SOCKS para cada gestor de colas del clúster que abarque Internet y habilitando MQIPT para que actúe como proxy SOCKS.

En el diagrama siguiente, NEWYORK y CHICAGO están en un clúster llamado HOME y ambos contienen repositorios completos. NEWYORK, LONDON y PARIS están en otro clúster llamado INVENTORY. Tenga en cuenta que CHICAGO no tiene que estar habilitado para SOCKS ya que está en un clúster que no necesita un MQIPT.

Cada gestor de colas del clúster INVENTORY está realmente "oculto" detrás de un MQIPT. Puesto que el gestor de colas está habilitada para SOCKS, cuando se inicia un canal de clúster emisor, la solicitud se envía a su destino, utilizando MQIPT que actúa como un proxy SOCKS. Por regla general, el CONNAME en un canal de clúster receptor se utiliza para identificar el gestor de colas local, pero cuando se utiliza con MQIPT, el CONNAME debe identificar el MQIPT local y su puerto de escucha entrante. En el diagrama siguiente, todas las direcciones de puerto de escucha entrante son 1414 y las direcciones de puerto de escucha saliente son 1415.

Existen dos formas para ejecutar un gestor de colas habilitado para SOCKS. La primera es habilitar para SOCKS a todo el sistema donde se va a ejecutar el gestor de colas. El segundo es habilitar para SOCKS solo el gestor de colas. Utilizando cualquiera de los dos métodos, debe configurar el cliente SOCKS para que solo realice conexiones remotas utilizando MQIPT como el proxy SOCKS e inhabilitar la autenticación de usuario. Hay una serie de productos en el mercado para conseguir el soporte de SOCKS. Debe elegir uno que dé soporte al protocolo SOCKS V5.

Consulte Configuración del soporte de clúster de MQIPT para ver un ejemplo de cómo configurar una red de clúster.



Soporte de SSL/TLS en MQIPT

Se pueden utilizar los sockets seguros para garantizar la privacidad, integridad y autenticación de las comunicaciones.

Privacidad de comunicación
La conexión se puede convertir en privada. Los datos que se van a intercambiar entre el cliente y el servidor se pueden cifrar y solo pueden entender los datos el emisor y el receptor. Esto significa que la información privada como, por ejemplo, números de tarjeta crédito, se puede transferir de forma segura.

Integridad de comunicación
La conexión es fiable. El transporte de mensajes incluye una comprobación de integridad de mensaje basada en una función hash segura.

Autenticación
El cliente puede autenticar el servidor y un servidor autenticado puede autenticar el cliente. Esto significa que se garantiza que la información se va a intercambiar solo entre las partes previstas.

El mecanismo de autenticación se basa en el intercambio de certificados digitales (certificados X.509v3).

Protocolos de sockets seguros

En MQIPT, los sockets seguros se proporcionan utilizando los protocolos Transport Layer Security (TLS) y Secure Sockets Layer (SSL). Los dos protocolos de sockets seguros son similares, pero no interoperan. En esta documentación, los términos SSL y TLS se utilizan indistintamente, a menos que se indique una diferencia específica.

MQIPT admite SSL 3.0, TLS 1.0, TLS 1.1, y TLS 1.2 proporcionados por el Java runtime environment (JRE) suministrado. A partir de IBM MQ 9.3.0, MQIPT también da soporte a TLS 1.3. La CipherSpec de IBM MQ del canal remoto determina qué protocolo utiliza MQIPT.

SSL 3.0, TLS 1.0y TLS 1.1 no son seguros y están inhabilitados de forma predeterminada en MQIPT. Si necesita utilizar alguno de estos protocolos inhabilitados, se pueden volver a habilitar siguiendo el procedimiento en "Habilitación de protocolos y suites de cifrado en desuso en MQIPT" en la página 1059.

Los protocolos SSL/TLS pueden utilizar algoritmos de firma digital diferentes para la autenticación de las partes de comunicación. Las operacionescriptográficas que se utilizan en SSL/TLS, el cifrado para la confidencialidad de datos y el hashing seguro para la integridad de los mensajes se basan en el uso compartido de claves secretas entre el cliente y el servidor. SSL/TLS proporciona distintos mecanismos de intercambio de claves que están permitidos para el uso compartido de claves secretas. SSL/TLS puede utilizar varios algoritmos para el cifrado y el hashing.

Habilitación de la modalidad FIPS en MQIPT

El componente criptográfico SSL/TLS del JRE contiene el proveedor de seguridad de `IBMJCEPlusFIPS` , que está certificado conforme con el estándar FIPS 140-2. Si desea utilizar sólo criptografía certificada por FIPS en MQIPT, habilite la modalidad FIPS en el proveedor `IBMJSSE2` estableciendo las siguientes propiedades del sistema Java cuando se inicie MQIPT :

- `com.ibm.jsse2.usefipsprovider=true`
- `com.ibm.jsse2.usefipsProviderName=IBMJCEPlusFIPS`

Puede establecer las propiedades del sistema Java cuando se inicia MQIPT utilizando la variable de entorno **`MQIPT_JVM_OPTIONS`** . Por ejemplo, en Linux, emita el mandato siguiente para establecer la variable de entorno, antes de emitir el mandato para iniciar MQIPT:

```
export MQIPT_JVM_OPTIONS="-Dcom.ibm.jsse2.usefipsprovider=true
-Dcom.ibm.jsse2.usefipsProviderName=IBMJCEPlusFIPS"
```

Para obtener más información sobre cómo habilitar la modalidad FIPS, consulte Habilitación de la modalidad FIPS en el proveedor IBMJSSE2.

Modalidad de puente SSL/TLS

Cuando una ruta tiene el SSLServer y el SSLClient establecidos, el MQIPT acepta una conexión segura SSL/TLS entrante y establece una segunda conexión segura SSL/TLS con otro MQIPT o con un gestor de colas de destino. La información del canal IBM MQ se descifra y se vuelve a cifrar entre estas dos conexiones SSL/TLS. El puente SSL/TLS también se conoce como *proxy de terminación SSL/TLS*.

IBM MQ da soporte al puente SSL/TLS utilizando MQIPT. Se ha observado que otros proxies de terminación SSL/TLS con IBM MQ provocan conexiones rotas si el proxy combina o reconstruye registros SSL/TLS con tamaños diferentes a los enviados por IBM MQ. Esto se debe a una interacción entre la forma en que los gestores de colas asignan y gestionan la memoria para los datos de red de IBM MQ entrantes y la forma en que los datos de red de IBM MQ se empaquetan en registros SSL/TLS.

El MQIPT conserva el empaquetado de datos de red de IBM MQ en registros SSL/TLS sin dividirlos ni combinarlos. Si otros puentes SSL/TLS no conservan los registros SSL/TLS exactamente, pueden hacer que los canales IBM MQ fallen con mensajes de error:

```
AMQ9638: SSL communications error for channel
AMQ9208: Error on receive from host
```

Modalidad de proxy SSL/TLS

Una ruta de MQIPT se puede configurar en modalidad de proxy SSL/TLS como alternativa al puente SSL/TLS. En esta modalidad, la ruta sólo reenvía datos SSL/TLS entre dos puntos finales de IBM MQ ; no participa en el reconocimiento SSL/TLS y no requiere ningún certificado digital.

Puede utilizar la modalidad de proxy SSL/TLS en los casos en los que los canales IBM MQ que se comunican a través de MQIPT ya están configurados para la comunicación SSL/TLS y desea utilizar MQIPT para otra finalidad, como el direccionamiento de conexiones a través de cortafuegos o la restricción del conjunto de conexiones permitidas a través de una salida de seguridad. Al ejecutar en modalidad de proxy SSL/TLS, MQIPT comprueba que los paquetes SSL/TLS iniciales recibidos de una nueva conexión son válidos antes de reenviar los paquetes al destino.

IBM MQ da soporte a la modalidad de proxy SSL/TLS con MQIPT o cualquier otro proxy SSL/TLS

IBM MQ soporte de múltiples certificados con MQIPT

IBM MQ 8.0, y posteriormente, da soporte al uso de varios certificados en el mismo gestor de colas, utilizando una etiqueta de certificado por canal, especificada utilizando el atributo **CERTLABL** en la definición de canal. Los canales de entrada al gestor de colas (por ejemplo, conexión con servidor o receptor) se basan en detectar el nombre de canal utilizando la indicación de nombre de servidor (SNI) de TSL, a fin de presentar el certificado correcto del gestor de colas. Para obtener más información sobre la utilización de varios certificados en un gestor de colas, veaCómo IBM MQ proporciona la funcionalidad de varios certificados.

Si un canal se conecta al gestor de colas de destino a través de MQIPT, y la ruta MQIPT tiene establecidos **SSLServer** y **SSLClient** , hay dos sesiones TLS separadas entre los puntos finales. En versiones anteriores a IBM MQ 9.3.0, los datos SNI no fluyen a través de la interrupción de sesión. Esto impide que se utilice un certificado por canal en el gestor de colas de destino, para la conexión TLS entre MQIPT y el gestor de colas. Para utilizar un certificado por canal en el gestor de colas de destino, para una conexión TLS que pasa a través de MQIPT en una versión anterior a IBM MQ 9.3.0, la ruta MQIPT debe utilizar la modalidad de proxy SSL/TLS, que reenvía todos los flujos de control TLS intactos, incluido el nombre SNI.

A partir de IBM MQ 9.3.0, MQIPT se puede configurar utilizando la propiedad de ruta SSLClientOutboundSNI para establecer las conexiones SNI para TLS en un valor específico o para pasar por el SNI recibido en la conexión de entrada a la ruta. Para permitir que se utilicen certificados por canal en un gestor de colas de destino, la ruta debe configurarse para establecer la SNI en el nombre de canal IBM MQ o pasar a través del SNI recibido en la conexión de entrada a la ruta. Si MQIPT está configurado para pasar a través de SNI, el gestor de colas o el cliente que se conecta a MQIPT debe establecer la SNI en el nombre de canal.

Los certificados que se utilizan para las conexiones TLS terminadas o iniciadas por MQIPT se pueden configurar individualmente para cada ruta, por ejemplo utilizando las propiedades de ruta **SSLServerSiteLabel** y **SSLClientSiteLabel**.

CipherSuites soportadas por MQIPT

La Tabla 71 en la página 1037 muestra qué CipherSuites están soportadas por MQIPT y cuáles están habilitadas de forma predeterminada.

De forma predeterminada, solo está habilitado un subconjunto de CipherSuites. Las CipherSuites basadas en varios algoritmos que se consideran inseguros están inhabilitadas por el JRE. Si conoce los posibles riesgos, pero sigue necesitando utilizar una de estas CipherSuites, puede añadir soporte para una

CipherSuite inhabilitada siguiendo el procedimiento de “Habilitación de protocolos y suites de cifrado en desuso en MQIPT” en la página 1059.

Tabla 71. CipherSuites que puede utilizar con MQIPT

CipherSuite	**Habilitada de forma predeterminada**
CipherSuites for TLS 1.3	
TLS_AES_128_GCM_SHA256	Sí
TLS_AES_256_GCM_SHA384	Sí
TLS_CHACHA20_POLY1305_SHA256	Sí
CipherSuites for SSL 3.0, TLS 1.0, TLS 1.1 y TLS 1.2	
SSL_DH_anon_EXPORT_WITH_DES40_CBC_SHA	
SSL_DH_anon_EXPORT_WITH_RC4_40_MD5	
SSL_DH_anon_WITH_3DES_EDE_CBC_SHA	
SSL_DH_anon_WITH_AES_128_CBC_SHA	
SSL_DH_anon_WITH_AES_128_CBC_SHA256	
SSL_DH_anon_WITH_AES_128_GCM_SHA256	
SSL_DH_anon_WITH_AES_256_CBC_SHA	
SSL_DH_anon_WITH_AES_256_CBC_SHA256	
SSL_DH_anon_WITH_AES_256_GCM_SHA384	
SSL_DH_anon_WITH_DES_CBC_SHA	
SSL_DH_anon_WITH_RC4_128_MD5	
SSL_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA	
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA	
SSL_DHE_DSS_WITH_AES_128_CBC_SHA	Sí
SSL_DHE_DSS_WITH_AES_128_CBC_SHA256	Sí
SSL_DHE_DSS_WITH_AES_128_GCM_SHA256	Sí
SSL_DHE_DSS_WITH_AES_256_CBC_SHA	Sí
SSL_DHE_DSS_WITH_AES_256_CBC_SHA256	Sí
SSL_DHE_DSS_WITH_AES_256_GCM_SHA384	Sí
SSL_DHE_DSS_WITH_DES_CBC_SHA	
SSL_DHE_DSS_WITH_RC4_128_SHA	
SSL_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA	
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA	
SSL_DHE_RSA_WITH_AES_128_CBC_SHA	Sí
SSL_DHE_RSA_WITH_AES_128_CBC_SHA256	Sí
SSL_DHE_RSA_WITH_AES_128_GCM_SHA256	Sí

Tabla 71. CipherSuites que puede utilizar con MQIPT (continuación)	
CipherSuite	**Habilitada de forma predeterminada**
SSL_DHE_RSA_WITH_AES_256_CBC_SHA	Sí
SSL_DHE_RSA_WITH_AES_256_CBC_SHA256	Sí
SSL_DHE_RSA_WITH_AES_256_GCM_SHA384	Sí
SSL_DHE_RSA_WITH_DES_CBC_SHA	
SSL_ECDH_anon_WITH_3DES_EDE_CBC_SHA	
SSL_ECDH_anon_WITH_AES_128_CBC_SHA	
SSL_ECDH_anon_WITH_AES_256_CBC_SHA	
SSL_ECDH_anon_WITH_NULL_SHA	
SSL_ECDH_anon_WITH_RC4_128_SHA	
SSL_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA	
SSL_ECDH_ECDSA_WITH_AES_128_CBC_SHA	Sí
SSL_ECDH_ECDSA_WITH_AES_128_CBC_SHA256	Sí
SSL_ECDH_ECDSA_WITH_AES_128_GCM_SHA256	Sí
SSL_ECDH_ECDSA_WITH_AES_256_CBC_SHA	Sí
SSL_ECDH_ECDSA_WITH_AES_256_CBC_SHA384	Sí
SSL_ECDH_ECDSA_WITH_AES_256_GCM_SHA384	Sí
SSL_ECDH_ECDSA_WITH_NULL_SHA	
SSL_ECDH_ECDSA_WITH_RC4_128_SHA	
SSL_ECDH_RSA_WITH_3DES_EDE_CBC_SHA	
SSL_ECDH_RSA_WITH_AES_128_CBC_SHA	Sí
SSL_ECDH_RSA_WITH_AES_128_CBC_SHA256	Sí
SSL_ECDH_RSA_WITH_AES_128_GCM_SHA256	Sí
SSL_ECDH_RSA_WITH_AES_256_CBC_SHA	Sí
SSL_ECDH_RSA_WITH_AES_256_CBC_SHA384	Sí
SSL_ECDH_RSA_WITH_AES_256_GCM_SHA384	Sí
SSL_ECDH_RSA_WITH_NULL_SHA	
SSL_ECDH_RSA_WITH_RC4_128_SHA	
SSL_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA	
SSL_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	Sí
SSL_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	Sí
SSL_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	Sí
SSL_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	Sí
SSL_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	Sí

Tabla 71. CipherSuites que puede utilizar con MQIPT (continuación)	
CipherSuite	**Habilitada de forma predeterminada**
SSL_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	Sí
SSL_ECDHE_ECDSA_WITH_NULL_SHA	
SSL_ECDHE_ECDSA_WITH_RC4_128_SHA	
SSL_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA	
SSL_ECDHE_RSA_WITH_AES_128_CBC_SHA	Sí
SSL_ECDHE_RSA_WITH_AES_128_CBC_SHA256	Sí
SSL_ECDHE_RSA_WITH_AES_128_GCM_SHA256	Sí
SSL_ECDHE_RSA_WITH_AES_256_CBC_SHA	Sí
SSL_ECDHE_RSA_WITH_AES_256_CBC_SHA384	Sí
SSL_ECDHE_RSA_WITH_AES_256_GCM_SHA384	Sí
SSL_ECDHE_RSA_WITH_NULL_SHA	
SSL_ECDHE_RSA_WITH_RC4_128_SHA	
SSL_KRB5_EXPORT_WITH_DES_CBC_40_MD5	
SSL_KRB5_EXPORT_WITH_DES_CBC_40_SHA	
SSL_KRB5_EXPORT_WITH_RC4_40_MD5	
SSL_KRB5_EXPORT_WITH_RC4_40_SHA	
SSL_KRB5_WITH_3DES_EDE_CBC_MD5	
SSL_KRB5_WITH_3DES_EDE_CBC_SHA	
SSL_KRB5_WITH_DES_CBC_MD5	
SSL_KRB5_WITH_DES_CBC_SHA	
SSL_KRB5_WITH_RC4_128_MD5	
SSL_KRB5_WITH_RC4_128_SHA	
SSL_RSA_EXPORT_WITH_DES40_CBC_SHA	
SSL_RSA_EXPORT_WITH_RC4_40_MD5	
SSL_RSA_WITH_3DES_EDE_CBC_SHA	
SSL_RSA_WITH_AES_128_CBC_SHA	Sí
SSL_RSA_WITH_AES_128_CBC_SHA256	Sí
SSL_RSA_WITH_AES_128_GCM_SHA256	Sí
SSL_RSA_WITH_AES_256_CBC_SHA	Sí
SSL_RSA_WITH_AES_256_CBC_SHA256	Sí
SSL_RSA_WITH_AES_256_GCM_SHA384	Sí
SSL_RSA_WITH_DES_CBC_SHA	
SSL_RSA_WITH_NULL_MD5	

Tabla 71. CipherSuites que puede utilizar con MQIPT (continuación)	
CipherSuite	**Habilitada de forma predeterminada**
SSL_RSA_WITH_NULL_SHA	
SSL_RSA_WITH_NULL_SHA256	
SSL_RSA_WITH_RC4_128_MD5	Sí
SSL_RSA_WITH_RC4_128_SHA	
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	
TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256	
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	

CipherSpecs y CipherSuites de MQIPT

La Tabla 72 en la página 1041 muestra la relación entre las CipherSpecs soportadas por IBM MQ y las CipherSuites soportadas por MQIPT. La tabla también muestra la versión del protocolo que IBM MQ espera que utilice cada CipherSpec.

Cuando un gestor de colas o cliente de IBM MQ se comunica con MQIPT, a menos que esté utilizando la modalidad de proxy SSL en MQIPT, asegúrese de que la CipherSpec que utiliza IBM MQ coincide con la CipherSuite que utiliza MQIPT:

- Cuando MQIPT actúa como servidor TLS y IBM MQ se conecta como cliente TLS, la CipherSpec utilizada por IBM MQ debe corresponder a una CipherSuite seleccionada en la configuración de ruta de MQIPT .
- Cuando MQIPT actúa como cliente TLS y se conecta a un gestor de colas IBM MQ que actúa como servidor TLS, la MQIPT CipherSuite debe coincidir con la CipherSpec que está definida en el canal IBM MQ receptor.

Una IBM MQ CipherSpec determina de forma exclusiva tanto el algoritmo de cifrado como la versión del protocolo de socket seguro que se va a utilizar. Algunas IBM MQ CipherSpecs sólo difieren por versión de protocolo. Si se utiliza una de estas CipherSpecs , no es suficiente especificar sólo la CipherSuite en la configuración de MQIPT . El reconocimiento SSL/TLS negocia la versión de protocolo de sockets seguros más alta soportada por ambos lados y, a continuación, selecciona una CipherSuite del conjunto de cifrados habilitados mutuamente.

Por ejemplo, una ruta SSLClient con `SSLClientCipherSuites=SSL_RSA_WITH_3DES_EDE_CBC_SHA` puede negociar para utilizar TLS_RSA_WITH_3DES_EDE_CBC_SHA (TLS 1.0) o TRIPLE_DES_SHA_US (SSL 3.0) con el gestor de colas remoto. También es posible utilizar esta CipherSuite con TLS 1.2, pero IBM MQ no da soporte a esta CipherSuite con TLS 1.2. Por este motivo, las rutas SSLClient son particularmente propensas a provocar errores AMQ9616 o AMQ9631 en el gestor de colas.

Para evitar estos errores en las rutas SSLClient, establezca la propiedad de ruta **`SSLClientProtocols`** en el valor adecuado para la CipherSpec prevista. En algunos casos, también puede ser necesario restringir el conjunto de protocolos del lado del servidor utilizando la propiedad de ruta **`SSLServerProtocols`**. Utilice la versión de protocolo que se muestra en la tabla para determinar el valor correcto para estas propiedades de ruta.

Este problema afecta especialmente a las CipherSuites y CipherSpecs siguientes para las rutas SSLClient:

- SSL_RSA_WITH_3DES_EDE_CBC_SHA, que corresponde a:
  - SSL 3.0: MQ CipherSpec TRIPLE_DES_SHA_US
  - TLS 1.0: MQ CipherSpec TLS_RSA_WITH_3DES_EDE_CBC_SHA
- SSL_RSA_WITH_DES_CBC_SHA, que corresponde a:
  - SSL 3.0: MQ CipherSpec DES_SHA_EXPORT

  - TLS 1.0: MQ CipherSpec TLS_RSA_WITH_DES_CBC_SHA
- SSL_RSA_WITH_RC4_128_SHA, que corresponde a:
  - SSL 3.0: MQ CipherSpec RC4_SHA_US
  - TLS 1.2: MQ CipherSpec TLS_RSA_WITH_RC4_128_SHA256

Si utiliza una única ruta de MQIPT SSLClient para crear un túnel en varios canales IBM MQ que utilizan CipherSpecsdiferentes, asegúrese de que todos los canales estén configurados con una CipherSpec que utilice la misma versión de protocolo de sockets seguros y que establezca la propiedad de ruta **SSLClientProtocols** en este protocolo.

Para obtener más información sobre CipherSpecs de IBM MQ, consulte Habilitación de CipherSpecs.

Tabla 72. IBM MQ CipherSpecs que corresponden a MQIPT CipherSuites

IBM MQ CipherSpec	**MQIPT CipherSuite**	**Versión de protocolo**
DES_SHA_EXPORT	SSL_RSA_WITH_DES_CBC_SHA	SSLv3
DES_SHA_EXPORT1024	No disponible	No disponible
ECDHE_ECDSA_3DES_EDE_CBC_SHA256	SSL_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA	TLSv1.2
ECDHE_ECDSA_AES_128_CBC_SHA256	SSL_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	TLSv1.2
ECDHE_ECDSA_AES_128_GCM_SHA256	SSL_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	TLSv1.2
ECDHE_ECDSA_AES_256_CBC_SHA384	SSL_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	TLSv1.2
ECDHE_ECDSA_AES_256_GCM_SHA384	SSL_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	TLSv1.2
ECDHE_ECDSA_NULL_SHA256	SSL_ECDHE_ECDSA_WITH_NULL_SHA	TLSv1.2
ECDHE_ECDSA_RC4_128_SHA256	SSL_ECDHE_ECDSA_WITH_RC4_128_SHA	TLSv1.2
ECDHE_RSA_3DES_EDE_CBC_SHA256	SSL_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA	TLSv1.2
ECDHE_RSA_AES_128_CBC_SHA256	SSL_ECDHE_RSA_WITH_AES_128_CBC_SHA256	TLSv1.2
ECDHE_RSA_AES_128_GCM_SHA256	SSL_ECDHE_RSA_WITH_AES_128_GCM_SHA256	TLSv1.2
ECDHE_RSA_AES_256_CBC_SHA384	SSL_ECDHE_RSA_WITH_AES_256_CBC_SHA384	TLSv1.2
ECDHE_RSA_AES_256_GCM_SHA384	SSL_ECDHE_RSA_WITH_AES_256_GCM_SHA384	TLSv1.2
ECDHE_RSA_NULL_SHA256	SSL_ECDHE_RSA_WITH_NULL_SHA	TLSv1.2
ECDHE_RSA_RC4_128_SHA256	SSL_ECDHE_RSA_WITH_RC4_128_SHA	TLSv1.2
NULL_MD5	SSL_RSA_WITH_NULL_MD5	SSLv3
NULL_SHA	SSL_RSA_WITH_NULL_SHA	SSLv3
RC2_MD5_EXPORT	No disponible	No disponible

Tabla 72. IBM MQ CipherSpecs que corresponden a MQIPT CipherSuites (continuación)		
IBM MQ CipherSpec	**MQIPT CipherSuite**	**Versión de protocolo**
RC4_56_SHA_EXPORT1024	No disponible	No disponible
RC4_MD5_EXPORT	SSL_RSA_EXPORT_WITH_RC4_40_MD5	SSLv3
RC4_MD5_US	SSL_RSA_WITH_RC4_128_MD5	SSLv3
RC4_SHA_US	SSL_RSA_WITH_RC4_128_SHA	SSLv3
TLS_AES_128_GCM_SHA256	TLS_AES_128_GCM_SHA256	TLSv1.3
TLS_AES_256_GCM_SHA384	TLS_AES_256_GCM_SHA384	TLSv1.3
TLS_CHACHA20_POLY1305_SHA256	TLS_CHACHA20_POLY1305_SHA256	TLSv1.3
TLS_RSA_WITH_3DES_EDE_CBC_SHA	SSL_RSA_WITH_3DES_EDE_CBC_SHA	TLSv1
TLS_RSA_WITH_AES_128_CBC_SHA	SSL_RSA_WITH_AES_128_CBC_SHA	TLSv1
TLS_RSA_WITH_AES_128_CBC_SHA256	SSL_RSA_WITH_AES_128_CBC_SHA256	TLSv1.2
TLS_RSA_WITH_AES_128_GCM_SHA256	SSL_RSA_WITH_AES_128_GCM_SHA256	TLSv1.2
TLS_RSA_WITH_AES_256_CBC_SHA	SSL_RSA_WITH_AES_256_CBC_SHA	TLSv1
TLS_RSA_WITH_AES_256_CBC_SHA256	SSL_RSA_WITH_AES_256_CBC_SHA256	TLSv1.2
TLS_RSA_WITH_AES_256_GCM_SHA384	SSL_RSA_WITH_AES_256_GCM_SHA384	TLSv1.2
TLS_RSA_WITH_DES_CBC_SHA	SSL_RSA_WITH_DES_CBC_SHA	TLSv1
TLS_RSA_WITH_NULL_NULL	No disponible	No disponible
TLS_RSA_WITH_NULL_SHA256	SSL_RSA_WITH_NULL_SHA256	TLSv1.2
TLS_RSA_WITH_RC4_128_SHA256	SSL_RSA_WITH_RC4_128_SHA	TLSv1.2
TRIPLE_DES_SHA_US	SSL_RSA_WITH_3DES_EDE_CBC_SHA	SSLv3

Reconocimiento SSL/TLS en MQIPT

El proceso de reconocimiento SSL/TLS se produce durante la solicitud de conexión inicial entre el cliente y el servidor SSL/TLS, cuando se realiza la autenticación y negociación de CipherSuites.

Todas las CipherSuites SSL/TLS soportadas (consulte “Soporte de SSL/TLS en MQIPT” en la página 1034), con la excepción de las CipherSuites anónimas, requieren autenticación de servidor y permiten la autenticación de cliente; el servidor se puede configurar para solicitar la autenticación de cliente. Debe evitar el uso de las CipherSuites anónimas porque no proporcionan ningún tipo de garantía sobre la identidad del igual remoto. Es posible que un ataque del tipo de suplantación de identidad intercepte conexiones SSL/TLS anónimas sin su conocimiento. Utilice las CipherSuites anónimas solo en redes internas de confianza y solo si está preparado para aceptar el riesgo de intercepción de datos.

La autenticación de iguales de la comunicación en SSL/TLS se basa en la criptografía de clave pública y los certificados digitales X.509v3. Un sitio que se debe autenticar en el protocolo SSL/TLS requiere una clave privada y un certificado digital (que contiene la clave pública correspondiente junto con la información sobre la identidad del sitio), el tiempo de validez del certificado. Los certificados son firmados por una autoridad de certificación, los certificados de dichas autoridades se denominan

certificados de firmante. Un certificado seguido de uno o más certificados de firmante constituyen una cadena de certificados. Una cadena de certificados se caracteriza por el hecho de que, a partir del primer certificado (certificado de sitio), la firma de cada certificado de la cadena se puede verificar utilizando la clave pública incluida en el siguiente certificado de firmante.

Cuando se está estableciendo una conexión segura que requiere autenticación de servidor, el servidor envía al cliente una cadena de certificados para probar su identidad. El cliente SSL/TLS perseguirá el establecimiento de la conexión son el servidor solo si lo puede autenticar, por ejemplo, verifique la firma del certificado de sitio del servidor. Para poder verificar esa firma, el cliente SSL/TLS necesita confiar en el propio sitio del servidor o, al menos, en uno de los firmantes de la cadena de certificados proporcionada por el servidor. Los certificados de los sitios y firmantes de confianza se deben mantener en el lado del cliente para realizar esta verificación.

El cliente SSL/TLS inspecciona la cadena de certificado del servidor, empezando con el certificado de sitio. El cliente considera que la firma del certificado de sitio es válida en las circunstancias siguientes:

- El certificado de sitio está en el repositorio de los certificados de firmante o de sitio de confianza
- Un certificado de firmante de la cadena se puede validar basándose en su repositorio de certificados de firmante de confianza

En el último caso, el cliente SSL/TLS comprueba que la cadena de certificados está firmada correctamente, desde el certificado de firmante de confianza hasta el certificado de sitio del servidor. Cada certificado implicado en este proceso también se examina para comprobar que el formato y la fechas de validez son correctos. Si falla alguna de estas comprobaciones, la conexión con el servidor se rechaza. Tras verificar el certificado del servidor, el cliente utiliza la clave pública incorporada en dicho certificado en los pasos siguientes del protocolo SSL/TLS. La conexión SSL/TLS solo se puede establecer si el servidor tiene realmente la clave privada correspondiente.

La autenticación de cliente sigue el mismo procedimiento; si un servidor SSL/TLS requiere la autenticación del cliente, el cliente envía al servidor una cadena de certificado para probar su identidad. El servidor verifica la cadena basándose en su repositorio de los certificados de firmante y sitio de confianza. Tras verificar el certificado del cliente, el servidor utiliza la clave pública incorporada en dicho certificado en los pasos siguientes del protocolo SSL/TLS. La conexión SSL/TLS solo se puede establecer si el cliente tiene realmente la clave privada correspondiente.

Las versiones recientes de los protocolos TLS proporcionan comunicaciones alta seguridad (SSL y los protocolos TLS más antiguos no se consideran seguros). Sin embargo, el protocolo funciona basándose en la información proporcionada por la aplicación. Solo si dicha base de información también se mantiene segura se puede conseguir el objetivo general de la comunicación segura. Por ejemplo, si el repositorio de los certificados de firmante y de sitio de confianza se ve comprometido, podría establecer una conexión segura con un socio de comunicaciones muy inseguro.

Implementación de MQIPT de SSL/TLS

SSL 3.0, TLS 1.0, TLS 1.1, TLS 1.2 y TLS 1.3 se implementan con señales PKCS (Public Key Cryptography Standards) #12 almacenadas en archivos de conjunto de claves (con tipos de archivo `.p12` o `.pfx`), que contienen X509.V3 . MQIPT también puede utilizar almacenes de claves de hardware de cifrado que dan soporte al estándar de la interfaz de señal criptográfica PKCS#11. MQIPT utiliza el paquete de IBM Java Secure Socket Extension (JSSE).

MQIPT puede actuar como un cliente SSL/TLS o un servidor SSL/TLS en función de qué extremo inicia la conexión. El cliente inicia una conexión y el servidor acepta la solicitud de conexión. Es posible para una ruta de MQIPT actuar como cliente y, también, como servidor. En este caso, el uso de la característica de modalidad de proxy SSL/TLS normalmente proporciona un mejor rendimiento.

Cuando MQIPT se ha configurado para la modalidad de proxy SSL/TLS, solo envía datos de SSL/TLS entre los dos puntos finales; no participa en el reconocimiento de SSL/TLS y no requiere ningún certificado digital.

En versiones anteriores a IBM MQ 9.3.0, MQIPT no pasa los datos SNI (Indicación de nombre de servidor TLS) que se reciben en una conexión TLS de entrada a través de una conexión TLS de salida. Esto significa que los certificados por canal, especificados utilizando el atributo de canal **CERTLABL**, no se

pueden utilizar para las conexiones TLS entre MQIPT y el gestor de colas de destino. Para utilizar un certificado por canal en el gestor de colas de destino, para una conexión TLS que pasa a través de MQIPT en una versión anterior a IBM MQ 9.3.0, la ruta MQIPT debe utilizar la modalidad de proxy SSL/TLS, que reenvía todos los flujos de control TLS intactos, incluido el nombre SNI. A partir de IBM MQ 9.3.0, MQIPT se puede configurar para establecer la SNI para conexiones TLS en un valor específico o para pasar a través de la SNI recibida en la conexión de entrada a la ruta. Para obtener más información sobre la utilización de varios certificados en un gestor de colas con MQIPT, consulte "IBM MQ soporte de múltiples certificados con MQIPT" en la página 1036.

Cada ruta de MQIPT se puede configurar de forma independiente con su propio conjunto de propiedades de SSL/TLS. Consulte Propiedades de ruta de MQIPT para obtener más detalles.

Cifrado de una contraseña de conjunto de claves en MQIPT

Las contraseñas utilizadas por MQIPT para abrir un archivo de conjunto de claves, o para acceder al hardware criptográfico, deben estar cifradas con el mandato **mqiptPW** .

La contraseña cifrada se puede especificar en cualquiera de las propiedades siguientes:

- **SSLClientKeyRingPW**
- **SSLClientCAKeyRingPW**
- **SSLServerKeyRingPW**
- **SSLServerCAKeyRingPW**
- **SSLCommandPortKeyRingPW**

En versiones anteriores a IBM MQ 9.1.5, las contraseñas de conjunto de claves para que las utilice MQIPT se almacenan en archivos a los que hacen referencia las propiedades **SSL*KeyRingPW** .

Cifre las contraseñas de conjunto de claves para que las utilice MQIPT utilizando el mandato **mqiptPW** y establezca el valor de las propiedades **SSL*KeyRingPW** en la contraseña cifrada. MQIPT puede distinguir entre contraseñas cifradas y nombres de archivo en valores de propiedad para la compatibilidad con configuraciones creadas antes de IBM MQ 9.1.5.

Deprecated El método de cifrar contraseñas de conjunto de claves en versiones de MQIPT anteriores a IBM MQ 9.1.5 está en desuso, pero se puede seguir utilizando. Para mejorar la protección de las contraseñas de conjunto de claves, vuelva a cifrar las contraseñas de conjunto de claves cifradas utilizando el método en desuso, con el método de protección más reciente.

Para cifrar una contraseña de conjunto de claves para que la utilice MQIPT, siga los pasos de "Cifrado de contraseñas almacenadas en MQIPT" en la página 1071.

Selección de certificados de un archivo de conjunto de claves en MQIPT

Es posible tener más de un certificado personal almacenado en el mismo archivo de conjunto de claves o señal de hardware de cifrado. Las propiedades **SSLClientSite*** se pueden utilizar en el lado del cliente para seleccionar el certificado que se va a enviar al servidor para la autenticación y las propiedades **SSLServerSite*** se pueden utilizar en el lado del servidor para seleccionar el certificado que se va a enviar al cliente para la autenticación.

Utilizando estas propiedades, se puede seleccionar un certificado basándose en su nombre distinguido (DN). De forma alternativa, la etiqueta de certificado se puede utilizar para seleccionar un certificado utilizando las propiedades **SSLServerSiteLabel** y **SSLClientSiteLabel**.

Para seleccionar el certificado de servidor utilizado por el puerto de mandatos TLS, utilice la propiedad **SSLCommandPortSiteLabel** para especificar el nombre de etiqueta del certificado.

Valores de confianza en MQIPT

Un almacén de claves contiene un certificado personal que incluye el certificado de firmante o la cadena de certificados de firmante.

MQIPT utiliza dos tipos de almacenes de claves:

Almacén de claves de entidad emisora de certificados (CA)

Este almacén de claves contiene certificados CA de confianza que se utilizan para validar certificados que pertenecen a un igual remoto. Estos certificados de CA ayudan a determinar si el igual remoto es de confianza. MQIPT da soporte a los almacenes de claves de formato PKCS #12 y a los almacenes de claves de hardware criptográfico que dan soporte a la interfaz PKCS #11 , para almacenar certificados de CA.

Los almacenes de claves de CA de MQIPT se identifican mediante las propiedades de ruta **SSLClientCAKeyRing** y **SSLServerCAKeyRing** . El uso del hardware criptográfico para acceder a los certificados de CA se habilita estableciendo las propiedades **SSLClientCAKeyRingUseCryptoHardware** y **SSLServerCAKeyRingUseCryptoHardware** .

El almacén de claves de CA en el cliente SSL/TLS debe contener los certificados de CA de confianza que se utilizan para autenticar el certificado enviado desde el servidor. Si se configura una ruta de servidor SSL para la autenticación de cliente, el almacén de claves de CA en el lado del servidor SSL/TLS debe contener los certificados de CA de confianza que se utilizan para autenticar el certificado enviado desde el cliente.

Almacén de claves de certificado personal

Este almacén de claves contiene certificados personales que MQIPT utiliza para identificarse en un igual remoto. Cuando genere un certificado autofirmado o solicite un certificado firmado por CA, hágalo utilizando el almacén de claves de certificados personales.

MQIPT da soporte a los almacenes de claves de formato PKCS #12 y a los almacenes de claves de hardware criptográfico que dan soporte a la interfaz PKCS #11 , para almacenar certificados personales.

Los almacenes de claves de certificados personales se identifican mediante las propiedades de ruta **SSLClientKeyRing** y **SSLServerKeyRing** . El uso del hardware criptográfico para acceder a certificados personales se habilita estableciendo las propiedades **SSLClientKeyRingUseCryptoHardware** y **SSLServerKeyRingUseCryptoHardware** .

El almacén de claves en el lado del servidor SSL/TLS debe contener el certificado personal del servidor MQIPT . Si la autenticación de cliente es necesaria en una ruta de cliente SSL, el almacén de claves en el lado del cliente SSL/TLS debe contener el certificado personal del cliente.

Si necesita la autenticación de cliente, debe habilitar la propiedad **SSLServerAskClientAuth** en el lado del servidor. El almacén de claves del lado del cliente debe contener el certificado personal del cliente. El almacén de claves de MQIPT en el lado del servidor, que se identifica mediante la propiedad **SSLServerCAKeyRing** , debe contener los certificados de CA de confianza que se utilizan para autenticar el cliente.

Si no configura un almacén de claves de CA para una ruta, MQIPT busca certificados de CA en el almacén de claves de certificados personales, si hay uno configurado. Por ejemplo, si no se establece ningún valor para **SSLServerCAKeyRing**, MQIPT busca certificados de CA en el almacén de claves identificado por **SSLServerKeyRing**.

Como alternativa al uso de certificados firmados por una CA de confianza, puede utilizar certificados autofirmados. Puede encontrar un ejemplo de un certificado autofirmado en el almacén de claves de ejemplo `sslSample.pfx` que se proporciona con MQIPT en el subdirectorio `samples/ssl` . Para abrir los almacenes de claves PKCS#12 de ejemplo, debe utilizar la contraseña `mqiptSample`.

Los certificados autofirmados pueden ser útiles en los escenarios de prueba en los que debe garantizar la conectividad de SSL/TLS sin pagar a una CA para un certificado. Sin embargo, no utilice certificados autofirmados en entornos de producción. Para crear un certificado firmado por CA, consulte Creación de un archivo de conjunto de claves.

Puede utilizar el mandato V 9.4.0 V 9.4.0 **mqiptKeytool** para gestionar certificados digitales y almacenes de claves. Para obtener más información, consulte “Gestión de almacenes de claves de MQIPT” en la página 1049.

Proteja los almacenes de claves y los archivos de contraseñas utilizando las características de seguridad del sistema operativo para evitar el acceso no autorizado a los mismos.

Prueba de SSL/TLS en MQIPT

Ejemplos para ayudarle a probar una conexión SSL/TLS.

Consulte Iniciación a IBM MQ Internet Pass-Thru para obtener una descripción de varios escenarios. En particular, consulte las tareas siguientes:

- Autenticación de un servidor SSL/TLS
- Autenticación de un cliente SSL/TLS
- Ejecución de MQIPT en modalidad de proxy SSL/TLS
- Ejecución de MQIPT en modalidad de proxy SSL/TLS con un gestor de seguridad

Para probar que la configuración de SSL/TLS funciona correctamente, puede utilizar certificados autofirmados. Los certificados autofirmados son útiles en escenarios de prueba para que pueda probar la conectividad SSL/TLS sin pagar una entidad emisora de certificados (CA) por un certificado. Para obtener más información, consulte Creación de certificados de prueba.

No utilice ningún certificado autofirmado en entornos de producción. En su lugar, obtenga un certificado firmado por CA de una CA de confianza. Para crear un certificado firmado por CA, consulte Creación de un archivo de conjunto de claves.

Al crear o solicitar un certificado, debe tener en cuenta qué tipo de clave, tamaño de clave y algoritmo de firma digital son adecuados para sus necesidades de seguridad. Para obtener más información, consulte “Consideraciones sobre el certificado digital para MQIPT” en la página 1051.

Los certificados y las tecnologías de gestión de certificados están disponibles en varios proveedores de terceros.

Conjuntos de claves de ejemplo de MQIPT

Los siguientes archivos de conjunto de claves PKCS #12 de ejemplo se proporcionan con MQIPT en el subdirectorio `samples/ssl` para su comodidad durante las pruebas:

`sslSample.pfx`
Un conjunto de claves que contiene un certificado autofirmado de ejemplo.

`sslCASample.pfx`
Un conjunto de claves de certificado de CA de ejemplo.

Para acceder a estos archivos de conjunto de claves de ejemplo, utilice la contraseña `mqiptSample`.

El certificado autofirmado de ejemplo sólo se debe utilizar en un entorno de prueba, ya que las claves privadas del certificado están disponibles para todos los usuarios de MQIPT .

Mensajes de error SSL/TLS en MQIPT

Las anomalías de reconocimiento se registran en el registro de conexiones MQIPT en forma de excepciones JSSE.

Para obtener más información, consulte “Registros de conexión en MQIPT” en la página 1073. En la tabla siguiente sed describen las distintas excepciones, la causa probable y la acción correspondiente para resolver la anomalía.

Las excepciones de certificado normalmente se relacionan con los certificados en el extremo remoto de la conexión.

Cuando el error está relacionado con el certificado de un cliente o un gestor de colas de IBM MQ, el término *archivo de conjunto de claves* incluye el repositorio de claves de IBM MQ del socio remoto.

En MQIPT, los certificado de CA se almacenan en el archivo de conjunto de claves de CA, que se identifica mediante las propiedades de ruta **`SSLClientCAKeyRing`** y **`SSLServerCAKeyRing`**. Si las propiedades de ruta del conjunto de claves de CA no están establecidas, en el archivo de conjunto de claves personal correspondiente (referenciado por la propiedad **`SSLClientKeyRing`** o **`SSLServerKeyRing`**) se buscan certificados de CA en su lugar.

Excepción	CAUSE	Acción
CertificateException	El certificado no es de confianza porque está firmado por una CA que no está en el conjunto de claves de CA.	Compruebe que en el archivo de conjunto de claves de CA están presentes todos los certificados de CA necesarios. Utilice la herramienta de gestión de claves de IBM proporcionada con MQIPT para añadir los certificados de CA que faltan, teniendo cuidado de obtener una copia de cada certificado de CA de una fuente de confianza.
CertificateExpiredException	1. El certificado ha caducado: ha pasado su fecha **notAfter**. 2. El reloj del sistema se ha establecido de forma incorrecta.	1. Obtenga un nuevo certificado e insértelo en el archivo de conjunto de claves. Si el certificado pertenece a una entidad emisora de certificados, coloque el nuevo certificado en el archivo de conjunto de claves de CA. 2. Compruebe que el reloj del sistema UTC está establecido en la hora correcta.
CertificateNotYetValidException	1. El certificado se está utilizando de forma prematura: su fecha **notBefore** todavía no ha llegado. 2. El reloj del sistema se ha establecido de forma incorrecta.	1. Compruebe que el certificado se ha generado y firmado correctamente. Si la organización opera su propia CA, el reloj del sistema UTC para la CA podría ser incorrecto. 2. Compruebe que el reloj del sistema UTC está establecido en la hora correcta.
CertificateParsingException	1. El certificado contiene datos DER no válidos. 2. El certificado utiliza características DER no soportadas.	Asegúrese de que el certificado se ha generado correctamente y se puede visualizar en la herramienta de gestión de claves IBM proporcionada con MQIPT. Considere la posibilidad de obtener un nuevo certificado con menos extensiones de certificado.
CertificateRevokedException	La comprobación de revocación de certificados está habilitada y se ha descubierto que el certificado se va a revocar.	El certificado en cuestión no debe ser de confianza. Obtenga un certificado de sustitución y asegúrese de que el nuevo certificado y su clave privada están presentes en el archivo de conjunto de claves.

Excepción	CAUSE	Acción
CertPathBuilderException	La cadena de certificados no ha sido firmada por una entidad emisora de certificados reconocida.	1. Si está utilizando certificados firmados por CA, compruebe que todos los certificados de CA raíz y de CA intermedia están presentes en el archivo de conjunto de claves de CA. 2. Si está utilizando certificados autofirmados, asegúrese de que ha extraído una copia de la parte pública del certificado remoto y de añadirlo al archivo de conjunto de claves de CA. Evite utilizar certificados autofirmados en entornos de producción.
CertStoreException KeyStoreException	Se ha producido un error al leer un certificado de un conjunto de claves por una de las razones siguientes: 1. El archivo de conjunto de claves está dañado. 2. Falta el archivo de conjunto de claves. 3. La contraseña almacenada no coincide con la contraseña del archivo de claves. 4. Si la ruta está configurada para utilizar hardware de cifrado, MQIPT podría no conectarse al hardware de cifrado.	1. Asegúrese de que el archivo de conjunto de claves es legible y que todos los certificados se pueden visualizar con la herramienta de gestión de claves IBM. 2. Compruebe que todas las propiedades de ruta del conjunto de claves hacen referencia al nombre de archivo correcto. 3. Compruebe que la contraseña del archivo de claves almacenada es correcta. Utilice la herramienta **mqiptPW** para almacenar la contraseña correcta. 4. Si la ruta está configurada para utilizar hardware de cifrado, compruebe lo siguiente: • El archivo de propiedades de seguridad de Java especifica que se ha instalado el proveedor de seguridad IBMPKCS11Impl. • El archivo de propiedades de seguridad de Java contiene el nombre completo del archivo de configuración que se utiliza para inicializar el proveedor de seguridad IBMPKCS11Impl. • El archivo de configuración que se utiliza para inicializar el proveedor de seguridad IBMPKCS11Impl es válido.

Excepción	CAUSE	Acción
SSLException: no hay certificados disponibles o la clave se corresponde a las suites de cifrado SSL que están habilitadas.	Debe tener un certificado personal con el tipo correcto de clave para las CipherSuites que está utilizando. Por ejemplo, las CipherSuites cuyos nombres empiezan con SSL_ECDH_ECDSA_ requieren un certificado con una clave pública Elliptic Curve. Las CipherSuites utilizadas con más frecuencia requieren un certificado con una clave pública RSA.	Abra el archivo de conjunto de claves con la herramienta de gestión de claves IBM. Bajo la vista Certificados personales, seleccione cada certificado a vez y vea el certificado. Pulse **Ver detalles** y vaya hasta la sección Clave pública de asunto para ver el tipo de clave pública. A continuación, compruebe las propiedades de ruta de MQIPT **SSLClientCipherSuites** y **SSLServerCipherSuites** para asegurarse de que están habilitadas las CipherSuites apropiadas.
SSLException: no hay ninguna suite de cifrado en común SSLHandshakeException: no hay ninguna suite de cifrado en común	El reconocimiento no ha podido llegar a un acuerdo sobre una CipherSuite porque no hay ningún solapamiento entre los conjuntos de CipherSuites habilitadas en ambos extremos de la conexión. En particular, una conexión IBM MQ de salida solo habilita un único cifrado, por lo que las rutas de MQIPT son especialmente susceptibles de sufrir este error. Este error también se puede producir cuando se cumplen las tres condiciones siguientes: • No se ha especificado ninguna CipherSuite en la ruta • No se puede encontrar ningún certificado de sitio apto en el conjunto de claves configurado para la ruta. • Las CipherSuites anónimas están inhabilitadas	Compruebe la lista de CipherSuites habilitadas en las propiedades de ruta de MQIPT **SSLClientCipherSuites** y **SSLServerCipherSuites**. Considere habilitar CipherSuites adicionales. Consulte la tabla proporcionada para determinar las CipherSuites correctas para habilitar para cada valor de CipherSpec de canal de IBM MQ. Si no se especifica ninguna CipherSuite en la ruta, compruebe que las propiedades de la ruta del anillo de claves hacen referencia al archivo de conjunto de claves correcto y que el conjunto de claves contiene un certificado personal que MQIPT puede utilizar. Si la ruta se ha configurado para utilizar hardware de cifrado, compruebe que el atributo **tokenlabel** del archivo de configuración que se utiliza para inicializar el proveedor de seguridad IBMPKCS11Impl especifica la etiqueta de señal de dispositivo criptográfico correcta.

Gestión de almacenes de claves de MQIPT

V 9.4.0 V 9.4.0 Utilice el mandato **mqiptKeytool** para gestionar certificados en almacenes de claves que utiliza IBM MQ Internet Pass-Thru (MQIPT).

V 9.4.0 V 9.4.0

A partir de IBM MQ 9.4.0, este mandato sustituye al mandato **mqiptKeycmd** que se utiliza para gestionar certificados en versiones anteriores de MQIPT.

Formato de almacén de claves necesario para MQIPT

V 9.4.0 V 9.4.0 MQIPT da soporte a almacenes de claves que utilizan el formato de archivo PKCS #12 . Cuando utilice el mandato **mqiptKeytool** para gestionar el almacén de claves MQIPT , especifique el parámetro **-storetype pkcs12** para indicar que el almacén de claves utiliza el formato PKCS #12 .

MQIPT también puede acceder a certificados almacenados en hardware criptográfico que da soporte a la interfaz PKCS #11 . También se puede utilizar la interfaz para gestionar certificados en el hardware de PKCS #11. Para obtener más información, consulte "Utilización del hardware de cifrado PKCS #11 en MQIPT" en la página 1059.

Cifrado de la contraseña de almacén de claves para MQIPT

Cifre la contraseña del almacén de claves en un formato que MQIPT pueda utilizar para acceder al archivo. Para obtener más información, consulte "Cifrado de una contraseña de conjunto de claves en MQIPT" en la página 1044.

El recurso de archivo de ocultación al que IBM MQ da soporte no está soportado por MQIPT. Utilice el mandato **mqiptPW** para cifrar la contraseña del almacén de claves en lugar de utilizar un archivo de ocultación.

Ejemplos

V 9.4.0 V 9.4.0

Los ejemplos siguientes muestran cómo se utiliza el mandato **mqiptKeytool** para gestionar certificados en un almacén de claves de MQIPT .

- El mandato siguiente crea un certificado personal autofirmado para fines de prueba:

```
mqiptKeytool -genkeypair -keystore key.p12 -storetype pkcs12 -storepass password
             -alias mqipt -dname "CN=Test Certificate,OU=Sales,O=Example,C=US"
             -keyalg RSA -keysize 2048 -sigalg SHA256withRSA
```

  El mandato crea un certificado digital con una clave pública RSA de 2048-bits y una firma digital que utiliza RSA con el algoritmo hash SHA-256. El certificado y sus claves públicas y privadas asociadas se almacenan en un almacén de claves de formato PKCS #12 denominado `key.p12`. El archivo de almacén de claves se crea si no existe.

  Cuando cree un certificado, elija un algoritmo de cifrado de clave pública, un tamaño de clave y un algoritmo de firma digital que sean adecuados para las necesidades de seguridad de su organización. Para obtener más información, consulte "Consideraciones sobre el certificado digital para MQIPT" en la página 1051.

  Este ejemplo utiliza un certificado firmado automáticamente que es adecuado para fines de prueba. En un entorno de producción, utilice en su lugar un certificado firmado por una entidad emisora de certificados.

- El mandato siguiente crea una solicitud de certificado para un certificado firmado por CA que se utilizará para fines de producción:

```
mqiptKeytool -certreq -keystore key.p12 -storetype pkcs12 -storepass password
             -alias mqipt -file cert.req
```

  El mandato crea una solicitud de firma de certificado (CSR) en el formato PKCS #10 . El CSR se puede enviar a una entidad emisora de certificados para solicitar un certificado firmado por CA. El par de claves pública y privada con el alias `mqipt` debe crearse antes de emitir este mandato, emitiendo el mandato **mqiptKeytool -genkeypair** en el ejemplo anterior.

- El mandato siguiente recibe el archivo de certificado personal firmado por CA que se denomina `cert.crt` en el almacén de claves:

```
mqiptKeytool -importcert -keystore key.p12 -storetype pkcs12 -storepass password -file
cert.crt
```

Importe el certificado de CA de la CA que ha firmado el certificado personal en el almacén de claves emitiendo el mandato siguiente:

```
mqiptKeytool -importcert -keystore key.p12 -storetype pkcs12 -storepass password
             -file ca.crt -alias rootCA
```

Consideraciones sobre el certificado digital para MQIPT

Los puntos a tener en cuenta incluyen el tamaño de clave de certificado, la selección de un algoritmo de firma digital adecuado para el certificado y el certificado digital y la compatibilidad con CipherSuite .

Consideraciones sobre el tamaño de clave de certificado para MQIPT

El tamaño de la clave pública depende de la política de seguridad de la organización y del algoritmo de cifrado utilizado. En general, los tamaños de clave más grandes son más seguros. La tabla siguiente lista los tamaños mínimos de clave que debe utilizar:

Algoritmo	Tamaño mínimo de clave (bits)
Elliptic Curve	256
RSA	2048

V 9.4.0 V 9.4.0 Especifique el tamaño de clave del certificado utilizando el parámetro **-keysize** cuando cree un certificado con el mandato **mqiptKeytool** .

Selección de un algoritmo de firma digital de certificado apropiado

Para evitar la falsificación de los certificados digitales, es importante utilizar un algoritmo de firma digital firme. Cuando cree o solicite un certificado, tenga cuidado de seleccionar un algoritmo correcto.

Debe evitar utilizar algoritmos de firma digital antiguos basados en MD5 o SHA-1 , ya que estos algoritmos no se consideran seguros. Si es posible, utilice uno de los algoritmos de firma digital con base SHA-2 más nuevo como, por ejemplo, SHA-256 con RSA (SHA256WithRSA).

V 9.4.0 V 9.4.0 Especifique el algoritmo de firma utilizando el parámetro **-sigalg** cuando cree un certificado con el mandato **mqiptKeytool** .

Compatibilidad de certificado digital y CipherSuite en MQIPT

No todas las CipherSuites se pueden utilizar con todos los certificados digitales. Existen distintos tipos de CipherSuite, agrupadas por su prefijo de nombre de CipherSuite. Cada tipo de CipherSuite impone distintas restricciones sobre el tipo de certificado digital que se puede utilizar. Estas restricciones se aplican a todas las conexiones SSL/TLS de MQIPT, pero resultan especialmente relevantes para los usuarios del cifrado Elliptic Curve. Durante un reconocimiento de socket seguro, MQIPT selecciona automáticamente un certificado personal para identificarse a sí mismo que sea adecuado para la CipherSuitenegociada. En la mayoría de los casos, MQIPT interopera automáticamente con el igual remoto. Sin embargo, en algunos casos de ejemplo es posible que necesite utilizar una CipherSuite MQIPT específica para interoperar con un sistema IBM MQ remoto.

V 9.4.0 V 9.4.0 El mandato **mqiptKeytool** se puede utilizar para crear certificados con claves públicas DSA, RSA y Elliptic Curve. Consulte a su entidad emisora de certificados para obtener consejos sobre la creación de otros tipos de certificado.

El tipo de certificado digital para utilizar depende del tipo de CipherSuite que está utilizando:

- Las CipherSuites con nombres que empiezan con SSL_ECDH_ECDSA_ y SSL_ECDHE_ECDSA_ requieren un certificado digital con una clave pública Elliptic Curve.
- Las CipherSuites con nombres que contienen *anon* son anónimas; no requieren ningún certificado digital para identificar el igual remoto. Dichas CipherSuites pueden evitar las sobrecargas de la gestión

del ciclo de vida de los certificados en las redes en las que se utilizan medios alternativos de autenticación pero, en general, evite su uso debido a la falta de autenticación.

- Otras CipherSuites requieren un certificado digital con una clave pública RSA.

Salida de certificado en MQIPT

La finalidad de una salida de certificado es validar un certificado del interlocutor SSL/TLS recibido por MQIPT.

Puede configurar la ruta MQIPT para que actúe como un cliente SSL/TLS cuando realiza una nueva conexión y para que actúe como un servidor SSL/TLS cuando reciba una solicitud de conexión. Durante el proceso de reconocimiento SSL/TLS, un cliente SSL/TLS recibe un certificado del interlocutor del servidor y el certificado se puede utilizar para autenticar el servidor. Un servidor SSL/TLS también puede recibir un certificado del interlocutor del cliente y el certificado se puede utilizar para autenticar el cliente.

Se llama a la salida de certificado cuando MQIPT recibe un certificado del interlocutor, lo que le permite realizar una validación adicional. Las excepciones capturadas por la salida son capturadas por MQIPT y se termina la solicitud de conexión. Por lo tanto, es un procedimiento recomendado para la salida capturar todas las excepciones y volver a pasar un código de retorno apropiado a MQIPT.

Se proporciona un ejemplo para mostrar que se puede implementar una salida de certificado; para obtener más información, consulte Utilización de una salida de certificado para autenticar un servidor SSL/TLS.

Nota: MQIPT se ejecuta en una sola Java Virtual Machine, así que una salida de certificado definida por usuarios podría poner en peligro el funcionamiento normal de MQIPT de una de estas formas:

- Afectar a los recursos del sistema
- Generar cuellos de botella
- Degradar el rendimiento

Debe probar los efectos de la salida de certificado ampliamente, antes de implementarla en un entorno de producción.

La clase com.ibm.mq.ipt.exit.CertificateExit en MQIPT

La clase com.ibm.mq.ipt.exit.CertificateExit es una clase abstracta que debe ser implementada por la clase definida con la propiedad SSLExitName .

La clase contiene las implementaciones predeterminadas para ejecutar la salida y algunos métodos públicos que puede alterar temporalmente, si lo desea, de acuerdo con sus requisitos. La lista completa de métodos soportados es la siguiente:

Métodos

public int init(IPTTrace)

MQIPT llama al método init cuando MQIPT carga la salida y se puede implementar para realizar cualquier inicialización de la salida; por ejemplo, la carga de datos que se utiliza durante el proceso de validación. La implementación predeterminada no hace nada.

public int refresh(IPTTrace)

El método de renovación se implementa para realizar una renovación de los datos; por ejemplo, la recarga de cualquier dato para el disco que se utiliza durante el proceso de validación. Se llama a este método cuando el administrador MQIPT ha emitido un mandato de renovación. La implementación predeterminada no hace nada.

public void close(IPTTrace)

El método de cierre se implementa para realizar cualquier tarea de mantenimiento cuando la ruta está a punto de detenerse o cuando MQIPT se está cerrando. La implementación predeterminada no hace nada.

public CertificateExitResponse validate(IPTTrace)

Se llama al método de validación para realizar la validación del certificado de igual. El objeto de retorno se puede utilizar para volver a pasar información a MQIPT; por ejemplo, un código de retorno y se puede añadir algún texto al registro de conexiones. La implementación predeterminada devuelve una CertificateExitResponse con CertificateExitResponse.OK.

Métodos soportados para obtener propiedades:

public int getListenerPort()
recupera el puerto de escucha de ruta - según lo definido por la propiedad ListenerPort

public String getDestination()
recupera la dirección de destino - según lo definido por la propiedad Destination

public int getDestinationPort()
recupera la dirección del puerto de escucha de destino - según lo definido por la propiedad DestinationPort

public String getClientIPAddress()
recupera la dirección IP del cliente que realiza la solicitud de conexión

public int getClientPortAddress()
recupera la dirección de puerto utilizada por el cliente que realiza la solicitud de la conexión

public boolean isSSLClient()
se utiliza para determinar si se está llamando a la salida como un cliente SSL/TLS o un servidor SSL/TLS. Si esto devuelve el valor true, la salida está en el lado del cliente de la conexión, validando el certificado obtenido del servidor. Si esto devuelve el valor false, la salida está en el lado del servidor de la conexión, validando el certificado enviado por el cliente. Es válido para que una ruta actúe tanto como servidor como cliente SSL/TLS, descifrando y volviendo a cifrar el tráfico. En esta situación, aunque hay una sola clase de salida, algunas instancias de la clase se llamarán como cliente y algunas como servidores. Puede utilizar isSSLClient para determinar la situación para una instancia determinada.

public int getConnThreadID()
se utiliza para recuperar el ID de la hebra de trabajador que está manejando la solicitud de conexión, que puede ser útil para la depuración.

public String getChannelName()
recupera el nombre del canal IBM MQ que se utiliza en la solicitud de conexión. Esto solo es disponible cuando la solicitud de entrada no está utilizando SSL/TLS y MQIPT está actuando como cliente SSL/TLS.

public String getQMName()
recupera el nombre del gestor de colas IBM MQ utilizado en la solicitud de conexión. Esto solo está disponible cuando la solicitud de cliente no está utilizando SSL/TLS y MQIPT está actuando como cliente SSL/TLS.

public boolean getTimedout()
es utilizado por la salida para determinar si el tiempo de espera ha caducado.

public IPTCertificate getCertificate()
recupera el certificado SSL/TLS que se debe validar.

public String getExitData()
recupera los datos de salida, tal como están definidos por la propiedad SSLExitData.

public String getExitName()
recupera el nombre de la salida, tal como está definida mediante la propiedad SSLExitName.

La clase com.ibm.mq.ipt.exit.CertificateExitResponse en MQIPT

Esta clase se utiliza para volver a pasar información a MQIPT después de que se haya validado un certificado.

Constructores

public CertificateExitResponse(*int rc*, string message)

Este constructor se puede utilizar para volver a pasar un código de retorno y algún texto de mensaje. Los códigos de razón posibles son:

- ExitRc.OK
- ExitRc.VALIDATE_ERROR
- ExitRc.VALIDATE_REJECTED

public CertificateExitResponse(*int rc*)

Este constructor se puede utilizar para volver a pasar un código de retorno, sin ningún texto de mensaje. Los códigos de razón posibles son:

- ExitRc.OK
- ExitRc.VALIDATE_ERROR
- ExitRc.VALIDATE_REJECTED

public CertificateExitResponse()

Este constructor se puede utilizar para volver a pasar un código de retorno ExitRc.OK, si ningún tipo de mensaje.

Métodos

public String getVersion()

Este método devuelve la versión de esta clase.

public String toString

Este método devolverá una representación de serie de la respuesta, por ejemplo, "Código de razón: 4, Mensaje: comprobación de CRL fallida.

La clase com.ibm.mq.ipt.exit.IPTCertificate en MQIPT

Esta clase contiene el certificado SSL/TLS que se va a validar.

Métodos

public int getVersion()

Este método devuelve la versión de esta clase.

public byte [] getDerEncoding()

Este método devuelve la codificación ASN.1/DER del certificado X.509, o NULL si hay un error.

public byte [] getPemEncoding()

Este método devuelve la codificación PEM (BASE64) del certificado X.509, o NULL si hay un error.

public String getLabel()

Este método devuelve la etiqueta de certificado, o NULL si hay un error.

public String getName()

Este método devuelve el nombre distinguido del certificado, o NULL si no está disponible. Por ejemplo:

```
CN=Test Queue Manager,OU=Sales,O=Example,L=London,C=GB
```

public String getIssuerName()

Este método devuelve el nombre distinguido del emisor del certificado o NULL si no está disponible. Por ejemplo:

```
CN=Certificate Authority,OU=Security,O=Example,L=New York,C=US
```

public IPTCertificate getSigner()

Este método devuelve el certificado de firmante, o NULL si no está disponible. Para un certificado firmado automáticamente, devolverá una referencia a sí mismo.

public String toString()

Este método devuelve una representación de serie del certificado.

La clase com.ibm.mq.ipt.exit.IPTTrace en MQIPT

Las funciones de rastreo de MQIPT proporcionan llamadas de entrada y salida, que se pueden utilizar al entrar y salir de un método. También hay varias llamadas de datos para rastrear información útil.

Métodos

public void entry(String *fid*)

Donde *fid* se utiliza para identificar dónde se ha realizado la llamada, por ejemplo, la clase y el nombre del método.

Este método escribe una entrada en el archivo de salida de rastreo con el nivel apropiado de indentación para registrar el punto en el que el flujo de control entra en un método. Esta llamada es opcional, pero si se utiliza, también se debe utilizar una llamada coincidente a "exit(String)" dentro del mismo método.

public void exit(String *fid*)

Donde *fid* se utiliza para identificar dónde se ha realizado la llamada, por ejemplo, la clase y el nombre del método.

Este método escribe una salida en el archivo de salida de rastreo con el nivel apropiado de indentación para registrar el punto en el cual el flujo de control deja un método. Este método solo se utiliza cuando se ha utilizado previamente una llamada a "entry(String)" dentro del mismo método.

public void exit(String *fid*, int *rc*)

Donde *fid* se utiliza para identificar dónde se ha realizado la llamada, por ejemplo, la clase y el nombre del método, y *rc* es el código de retorno numérico del método. Este método de rastreo se debe utilizar para registrar la salida de los métodos que devuelven un entero.

Este método escribe una salida en el archivo de salida de rastreo con el nivel de indentación apropiado para registrar el punto en el que el flujo de control abandona un método y el código de retorno numérico de dicho método. Este método solo se utiliza cuando se ha utilizado previamente una llamada a "entry(String)" dentro del mismo método.

public void exit(String *fid*, boolean *rc*)

Donde *fid* se utiliza para identificar dónde se ha realizado la llamada, por ejemplo la clase y el nombre de método, y *rc* es el código de retorno booleano que procede del método. Este método de rastreo se debe utilizar para registrar la salida de los métodos que devuelven un booleano.

Este método escribe una salida en el archivo de salida de rastreo con el nivel de indentación apropiado para registrar el punto en el que el flujo de control abandona un método y el código de retorno booleano que procede de dicho método. Este método solo se utiliza cuando se ha utilizado previamente una llamada a "entry(String)" dentro del mismo método.

public void data(String *fid*, String data)

Donde *fid* se utiliza para identificar dónde se ha realizado la llamada, por ejemplo, la clase y el nombre del método.

Este método escribe algunos datos de serie en el archivo de salida de rastreo.

public void data(String *fid*, int data)

Donde *fid* se utiliza para identificar dónde se ha realizado la llamada, por ejemplo, la clase y el nombre del método.

Este método escribe algunos datos enteros en el archivo de salida de rastreo.

public void data(String *fid*, byte[])

Donde *fid* se utiliza para identificar dónde se ha realizado la llamada, por ejemplo, la clase y el nombre del método.

Este método escribe algunos datos binarios en el archivo de salida de rastreo.

Rastreo de ejemplo

Para ayudar a diagnosticar problemas en una salida, puede utilizar el mismo recurso de rastreo que MQIPT, de forma alternativa, puede implementar sus propias funciones de rastreo. Si decide utilizar las funciones de rastreo de MQIPT, existen llamadas de entrada y de salida, que se pueden utilizar en una entrada a y en una salida de un método. También hay varias llamadas de datos para rastrear información útil, tal como se muestra en el ejemplo siguiente.

```
/**
* This method is called to initialize the exit (for example, for
* loading validation information) and place itself in a ready
* state to validate connection requests.
*/
 public int init(IPTTrace t) {
 final String fid = "MyExit.init";

 // Trace entry into this method
 t.entry(fid);

 // Trace useful information
 t.data(fid, "Starting exit - MQIPT version " + getVersion());

 // Perform initialization and load any data
 t.data(fid, "Ready for work");

 // Trace exit from this method
 t.exit(fid);

 return ExitRc.OK;
}
```

Este método genera un rastreo con el formato que se muestra en el ejemplo siguiente:

```
16:36:48.625    14   5000-1s    ------{ ConnectionThread.setCertificateExit()
16:36:48.625    14   5000-1s     Creating instance of certificate exit
16:36:48.625    14   5000-1s     Calling init() of certificate exit
16:36:48.625    14   5000-1s    -------{ MyExit.init()
16:36:48.625    14   5000-1s     Starting exit - MQIPT version 2.1.0.0
16:36:48.625    14   5000-1s    Ready for work
16:36:48.625    14   5000-1s    -------} MyExit.init() rc=0
16:36:48.625    14   5000-1s    ------} ConnectionThread.setCertificateExit() rc=0
```

Códigos de retorno de salida de certificado en MQIPT

Los códigos de retorno que MQIPT reconoce al llamar a una salida de certificado en una serie de situaciones diferentes

Los códigos de retorno siguientes son reconocidos por MQIPT al llamar a una salida de certificado en las situaciones siguientes:

Código de retorno	Descripción	init	Validate	Icono Renovar
ExitRc.OK	La solicitud se ha completado correctamente.	sí	sí	sí
ExitRc.INIT_ERROR	La solicitud de inicialización ha fallado, la ruta se inhabilitará.	sí		
ExitRc.REFRESH_ERROR	Ha fallado la solicitud de renovación, la ruta se inhabilitará.			sí
ExitRc.VALIDATE_ERROR	El proceso de validación ha fallado, se ha rechazado la solicitud de conexión.		sí	
ExitRc.VALIDATE_REJECTED	La solicitud de validación se ha rechazado, se ha rechazado la solicitud de conexión.		sí	

LDAP y CRL en MQIPT

MQIPT admite el uso de un servidor LDAP (Lightweight Directory Access Protocol) para realizar la autenticación de la lista de revocación de certificados (CRL) en un certificado digital.

El soporte de LDAP se ha implementado de una forma similar a la de IBM MQ, ya que se puede utilizar el mismo servidor LDAP para ambos, IBM MQ y MQIPT.

Durante el reconocimiento SSL/TLS, los participantes de la comunicación se autentican entre sí con certificados digitales. La autenticación puede incluir una comprobación de que el certificado recibido continúa siendo fiable. Las autoridades de certificación (CA) revocan los certificados por distintos motivos, entre los que se incluyen los siguientes:

- El propietario ha cambiado a una organización distinta.
- La clave privada ya no es secreta.

Las CA publican los certificados personales revocados en una Lista de revocación de certificados (CRL). Los certificados de CA que se han revocado se publican en una Lista de revocación de autorizaciones (ARL). Tenga en cuenta que las referencias posteriores a las CRL también se aplican a las ARL.

Para obtener más información sobre el uso de servidores LDAP con IBM MQ y sobre la gestión de las CRL y las ARL, consulte Trabajar con listas de revocación de certificados y listas de revocación de autorizaciones.

MQIPT puede dar soporte a un máximo de dos servidores LDAP en cada ruta. El primer servidor LDAP se trata como el servidor principal y el segundo servidor LDAP se mantiene como una copia de seguridad. El segundo servidor solo se utiliza si no se puede acceder al servidor principal. El servidor de seguridad debe ser una imagen duplicada del servidor principal.

El acceso a la información almacenada en un servidor LDAP se puede proteger con un ID de usuario y una contraseña utilizando las propiedades de contraseña e ID de usuario de LDAP. Las contraseñas del servidor LDAP se pueden cifrar en la configuración de MQIPT desde IBM MQ 9.1.5. Para obtener más información sobre el cifrado de contraseñas que utilizará MQIPT, consulte “Cifrado de contraseñas almacenadas en MQIPT” en la página 1071.

Cuando MQIPT carga una señal PKCS #12 de un archivo de conjunto de claves, los certificados de CA se comprueban para ver la validez de la CRL. Si el certificado de CA tiene una CRL adjunta, se comprueba para ver si ha caducado y, si es así, se recupera una CRL más nueva del servidor LDAP. Las CRL recuperadas se cargan en la señal actual y se adjuntan a su certificado de CA.

Si no hay ninguna entrada que coincida con la CA dada cuando se envía una consulta al servidor LDAP principal, se presupone que no hay ninguna CRL para esa CA y no se utiliza el servidor de copia de seguridad. Sin embargo, si no se puede alcanzar al servidor LDAP principal o no devuelve ninguna respuesta dentro de un marco de tiempo determinado, se utiliza el servidor de seguridad. Los errores del

servidor de copia de seguridad hacen que la conexión de cliente se termine. Esta acción se puede alterar temporalmente estableciendo la propiedad **LDAPIgnoreErrors** en `true`.

Las CRL recuperadas por MQIPT se conservan en una memoria caché y son compartidas por todas las conexiones en dicha ruta. Si una CRL almacenada en memoria caché ha caducado, la CRL se elimina de la memoria caché y se recupera una nueva del servidor LDAP. Si no está disponible una nueva CRL, la conexión se sigue rechazando.

Una CRL recuperada del servidor LDAP también se comprueba para ver si ha caducado y se muestra un mensaje de aviso (MQCPW001). La CRL caducada se sigue cargando en el sistema y se rechazan las solicitudes de conexión que hacen referencia a esa CRL. Debe sustituir la CRL caducada en el servidor con una actual.

La propiedad **LDAPCacheTimeout** se puede utilizar para controlar la frecuencia con la que se borra la memoria caché de CRL. El valor predeterminado es 1 día. Establecer este valor en `0` significa que las entradas de memoria caché no se borran hasta que se reinicia la ruta.

Se puede almacenar una CRL caducada en un archivo de conjunto de claves o en un servidor LDAP. Si no se ha emitido una nueva CRL, se rechazan las solicitudes de conexión adicionales. Puede ignorar las CRL caducadas habilitando la propiedad **IgnoreExpiredCRLs**.

Nota: Si habilita la propiedad **LDAPIgnoreErrors** o la propiedad **IgnoreExpiredCRLs**, se podría utilizar un certificado revocado para realizar una conexión SSL/TLS.

Propiedades de OU de nombre distinguido de certificado de varios valores en MQIPT

Puede emparejar varios valores de unidad organizativa (OU) en nombres distinguidos de certificado.

Ahora las propiedades de ruta siguientes dan soporte al emparejamiento de varios valores de OU:

- **SSLClientDN_OU**
- **SSLClientSiteDN_OU**
- **SSLServerDN_OU**
- **SSLServerSiteDN_OU**

Para emparejar varios valores de OU, utilice una coma como separador en el valor de propiedad de ruta. Por ejemplo:

```
SSLClientDN_OU=Sales, Europe
```

Esto empareja certificados con `OU=Sales` y `OU=Europe`. Los valores OU se emparejan en la misma secuencia con varios valores de OU en los filtros SSLPEER de IBM MQ.

No especifique la misma propiedad de ruta más de una vez en la sección `[route]`. La forma correcta de emparejar varios valores de OU es especificar la propiedad una vez, tal como se muestra en el ejemplo anterior. Si especifica el mismo atributo más de una vez en la misma sección `mqipt.conf`, el último valor entrará en vigor. Por ejemplo, las entradas siguientes solo darían como resultado el emparejamiento de `Europe` porque la segunda línea altera temporalmente la primera:

```
SSLClientDN_OU=Sales
SSLClientDN_OU=Europe
```

Si debe emparejar una coma literal dentro de un valor de OU, inserte una barra inclinada invertida (\) como carácter de escape inmediatamente antes de la coma. Por ejemplo:

```
SSLClientDN_OU=Sales\, Europe
```

Esto empareja un solo valor: `OU=Sales, Europe`. Una barra inclinada invertida que no va seguida inmediatamente de una coma se empareja con una barra inclinada invertida literal.

Si está actualizando desde un release anterior de MQIPT y se basa en la capacidad para emparejar comas en valores de OU, debe insertar caracteres de escape de barra inclinada invertida en las propiedades de ruta de OU para poder conservar el comportamiento anterior.

Deprecated Habilitación de protocolos y suites de cifrado en desuso en MQIPT

De forma predeterminada, los protocolos y los paquetes de cifrado de sockets seguros que no se consideran seguros están inhabilitados en el Java runtime environment (JRE) proporcionado con MQIPT. Estos protocolos y paquetes de cifrado en desuso se deben habilitar antes de poderse utilizar.

Acerca de esta tarea

Si conoce los posibles riesgos, pero sigue necesitando utilizar uno de los protocolos o uno de los paquetes de cifrado que no se consideran seguros en MQIPT, siga este procedimiento para habilitar el protocolo o el paquete de cifrado que necesite utilizar.

Nota: Los protocolos y los paquetes de cifrado en desuso no se pueden utilizar con el puerto de mandatos TLS.

Procedimiento

1. Edite el archivo `java.security`, que se encuentra en el directorio *`mqipt_path`*`/java/jre/lib/security`, donde *vía_acceso_mqipt* es la ubicación donde MQIPT está instalado.
2. Añada soporte al JRE para un protocolo o algoritmo eliminando la entrada correspondiente de la lista de algoritmos inhabilitados en la propiedad `jdk.tls.disabledAlgorithms`.
   - Para añadir soporte para un protocolo, elimine el protocolo de la lista de algoritmos inhabilitados. Por ejemplo, para añadir soporte para TLS 1.0, elimine TLSv1 de la lista.
   - Para añadir soporte para un paquete de cifrado, elimine los algoritmos correspondientes de la lista de algoritmos inhabilitados. Por ejemplo, para añadir soporte para la suite de cifrado de SSL_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA, elimine 3DES_EDE_CBC y DESede de la lista.
3. Para habilitar SSL 3.0 en el JRE, también debe establecer la propiedad del sistema `com.ibm.jsse2.disableSSLv3=false`.

   Si está iniciando MQIPT desde la línea de mandatos utilizando el mandato **mqipt**, puede establecer la propiedad utilizando la variable de entorno **MQIPT_JVM_OPTIONS**. Por ejemplo:

   ```
   set MQIPT_JVM_OPTIONS=-Dcom.ibm.jsse2.disableSSLv3=false
   ```

   Windows Si MQIPT se instala como un servicio de Windows, puede establecer la propiedad definiendo un valor de serie en el registro de Windows bajo la clave `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MQInternetPassThru`. El valor debe tener los atributos siguientes:

   Nombre
   MqiptJvmOptions

   Datos de valor
   -Dcom.ibm.jsse2.disableSSLv3=false
4. Para habilitar SSL 3.0, TLS 1.0 o TLS 1.1 en una ruta de MQIPT, añada el protocolo correspondiente a la propiedad de ruta **SSLServerProtocols** o **SSLClientProtocols**.
5. Reinicie MQIPT para que los cambios en las propiedades JRE entren en vigor.

Utilización del hardware de cifrado PKCS #11 en MQIPT

MQIPT puede acceder a los certificados digitales que se almacenan en hardware de cifrado que da soporte a la interfaz PKCS #11.

Antes de empezar

Antes de configurar MQIPT para utilizar hardware criptográfico, asegúrese de que la tarjeta criptográfica, el controlador de tarjeta y cualquier software de soporte asociado estén instalados y funcionen correctamente.

El soporte para el hardware de cifrado PKCS #11 en MQIPT es proporcionado por el IBM Java PKCS11 Cryptographic Provider (proveedor IBMPKCS11Impl). Para obtener más información sobre el proveedor IBMPKCS11Impl y la lista de tarjetas criptográficas a las que Java 8 da soporte, consulte Proveedor criptográficoIBM PKCS11.

Acerca de esta tarea

Puede almacenar los certificados personales y los certificados CA que MQIPT utiliza en un almacén de claves de hardware criptográfico. Sin embargo, como un dispositivo PKCS #11 normalmente no tiene suficiente espacio disponible para almacenar muchos certificados de firmante, es posible que desee utilizar un almacén de claves basado en archivos independiente para los certificados de CA.

Siga este procedimiento para configurar MQIPT para utilizar certificados en un almacén de claves de hardware de cifrado.

Nota:

- El uso del hardware de cifrado con MQIPT es una prestación de IBM MQ Advanced. Para utilizar esta prestación, también es necesario que el gestor de colas local que está conectado con la ruta de MQIPT tenga titularidad de IBM MQ Advanced, IBM MQ Appliance, IBM MQ Advanced for z/OS VUEo IBM MQ Advanced for z/OS .
- V 9.4.0 V 9.4.0 El mandato **mqiptKeytool** no da soporte actualmente al hardware criptográfico PKCS #11 . Este procedimiento utiliza el mandato **ikeycmd** que se proporciona con Java runtime environment (JRE) en su lugar para gestionar certificados en el hardware de cifrado. Asegúrese de que el mandato **ikeycmd** en el JRE en la vía de acceso de instalación de MQIPT se utiliza para ejecutar estos mandatos.

Procedimiento

1. Cree el archivo de configuración que se utiliza cuando se inicializa el proveedor IBMPKCS11Impl .

   Descargue los archivos de configuración de ejemplo para cada una de las tarjetas criptográficas de hardware soportadas por el proveedor IBMPKCS11Impl y configure un ejemplo para el sistema. Los ejemplos se pueden descargar desde el tema siguiente en IBM Documentation para Java: Archivo de configuración.

   El archivo de configuración es un archivo de texto y debe contener al menos los atributos siguientes:

   name
   El sufijo de nombre de la instancia del proveedor.

   library
   El nombre completo de la biblioteca PKCS #11 que se proporciona con el hardware de cifrado.

   tokenlabel
   La etiqueta de la señal del dispositivo criptográfico PKCS #11.

   Por ejemplo, el archivo de configuración puede contener las entradas siguientes:

   ```
   name = IPTPKCS11Provider
   library = /usr/lib64/pkcs11/PKCS11_API.so
   tokenlabel = icatoken
   ```

2. Edite el archivo de propiedades de seguridad de Java, `java.security`, que se encuentra en el subdirectorio `java/jre/lib/security` del directorio de instalación de MQIPT.

   a) Si todavía no está presente en el archivo, añada el proveedor de seguridad IBMPKCS11Impl . Por ejemplo, añadiendo la línea siguiente:

```
security.provider.12=com.ibm.crypto.pkcs11impl.provider.IBMPKCS11Impl
```

b) Añada el nombre completo del archivo de configuración después del nombre del proveedor. Por ejemplo, si el archivo de configuración que ha creado en el paso "1" en la página 1060 se denomina `/opt/mqipt/pkcs11.cfg`, añada esta vía de acceso a la misma línea que el proveedor de seguridad:

```
security.provider.12=com.ibm.crypto.pkcs11impl.provider.IBMPKCS11Impl /opt/mqipt/
pkcs11.cfg
```

3. Solicite un certificado personal para el hardware de cifrado.

   Especifique el mandato siguiente para crear una solicitud de certificado con el mandato **ikeycmd** :

```
MQIPT_INSTALL_DIRECTORY/java/jre/bin/ikeycmd -certreq -create
            -crypto module_name -tokenlabel hardware_token
            -pw password -label label -size key_size
            -sig_alg algorithm -dn distinguished_name -file filename
```

   donde:

   MQIPT_INSTALL_DIRECTORY
   es el directorio de instalación de MQIPT.

   -crypto *nombre_módulo*
   Especifica el nombre completo de la biblioteca PKCS #11 que se proporciona con el hardware de cifrado.

   -tokenlabel *etiqueta_señal*
   Especifica la etiqueta de señal del dispositivo criptográfico PKCS #11.

   -pw *contraseña*
   Especifica la contraseña para acceder al hardware de cifrado.

   -label *etiqueta*
   Especifica la etiqueta de certificado.

   -size *tamaño_clave*
   Especifica el tamaño de clave. El valor puede ser 512, 1024, 2048 o 4096.

   -sig_alg *algoritmo*
   Especifica el algoritmo de firma asimétrico que se utiliza para la creación del par de claves de la entrada. El valor puede ser `MD2_WITH_RSA`, `MD2WithRSA`, `MD5_WITH_RSA`, `MD5WithRSA`, `SHA1WithDSA`, `SHA1WithECDSA`, `SHA1WithRSA`, `SHA2/ECDSA`, `SHA224WithECDSA`, `SHA256_WITH_RSA`, `SHA256WithECDSA`, `SHA256WithRSA`, `SHA2WithECDSA`, `SHA3/ECDSA`, `SHA384_WITH_RSA`, `SHA384WithECDSA`, `SHA384WithRSA`, `SHA3WithECDSA`, `SHA5/ECDSA`, `SHA512_WITH_RSA`, `SHA512WithECDSA`, `SHA512WithRSA`, `SHA5WithECDSA`, `SHA_WITH_DSA`, `SHA_WITH_RSA` o `SHAWithDSA`. El valor predeterminado es `SHA256WithRSA`.

   -dn *nombre_distinguido*
   Especifica el nombre distinguido X.500 especificado entre comillas dobles.

   -file *nombrearchivo*
   Especifica el nombre de archivo para la solicitud de certificado.

4. Después de que la CA le envíe el certificado personal, añada el certificado de CA al almacén de claves de hardware criptográfico o a un archivo de almacén de claves de CA, si todavía no está presente.

   - V 9.4.0 V 9.4.0 Para añadir el certificado de CA a un archivo de almacén de claves de CA PKCS #12 , especifique el mandato siguiente:

```
mqiptKeytool -importcert -keystore filename -storetype pkcs12 -storepass password
             -alias label -file cert_filename
```

     donde:

-keystore ***nombre_archivo***
Especifica el nombre del archivo de almacén de claves de CA. El almacén de claves se crea si todavía no existe.

-storepass ***contraseña***
Especifica la contraseña del almacén de claves de CA.

-alias ***etiqueta***
Especifica la etiqueta del certificado de CA.

-file ***nombre_archivo_certificado***
Especifica el nombre del archivo que contiene el certificado de CA.

- Para añadir el certificado de CA al hardware de cifrado, especifique el mandato siguiente:

```
MQIPT_INSTALL_DIRECTORY/java/jre/bin/ikeycmd -cert -add
            -crypto module_name -tokenlabel hardware_token
            -pw password -label label -file cert_filename
```

donde:

MQIPT_INSTALL_DIRECTORY
es el directorio de instalación de MQIPT.

-crypto ***nombre_módulo***
Especifica el nombre completo de la biblioteca PKCS #11 que se proporciona con el hardware de cifrado.

-tokenlabel ***etiqueta_señal***
Especifica la etiqueta de señal del dispositivo criptográfico PKCS #11.

-pw ***contraseña***
Especifica la contraseña para acceder al hardware de cifrado.

-label ***etiqueta***
Especifica la etiqueta del certificado de CA.

-file ***nombre_archivo_certificado***
Especifica el nombre del archivo que contiene el certificado de CA.

5. Reciba el certificado personal, proporcionado por la CA, en el almacén de claves de hardware criptográfico.

   Especifique el mandato siguiente para añadir el certificado al almacén de claves de hardware de cifrado:

```
MQIPT_INSTALL_DIRECTORY/java/jre/bin/ikeycmd -cert -receive
            -crypto module_name -tokenlabel hardware_token
            -pw password -file filename
```

donde:

MQIPT_INSTALL_DIRECTORY
es el directorio de instalación de MQIPT.

-crypto ***nombre_módulo***
Especifica el nombre completo de la biblioteca PKCS #11 que se proporciona con el hardware de cifrado.

-tokenlabel ***etiqueta_señal***
Especifica la etiqueta de señal del dispositivo criptográfico PKCS #11.

-pw ***contraseña***
Especifica la contraseña para acceder al hardware de cifrado.

-label ***etiqueta***
Especifica la etiqueta de certificado.

-file ***nombre_archivo_certificado***
Especifica el nombre del archivo que contiene el certificado que se va a añadir.

Si el certificado de CA se almacena en un archivo de almacén de claves de CA, en lugar de en el hardware de cifrado, el mandato emite un aviso de que la cadena de certificados no se puede validar porque no se puede acceder al almacén de claves de CA cuando el certificado personal se añade al almacén de claves de hardware de cifrado.

6. Cifre la contraseña para acceder al hardware de cifrado utilizando el mandato **mqiptPW**.

   Escriba el mandato siguiente:

   ```
   mqiptPW -sf encryption_key_file
   ```

   donde *archivo_claves_cifrado* es el nombre de un archivo que contiene la clave de cifrado de contraseña para la instalación de MQIPT. No es necesario que especifique el parámetro **-sf** si la instalación de MQIPT utiliza la clave de cifrado de contraseña predeterminada. Introduzca la contraseña para acceder al hardware de cifrado para realizar el cifrado cuando se le solicite.

   Para obtener más información sobre el cifrado de contraseñas de almacén de claves, consulte "Cifrado de una contraseña de conjunto de claves en MQIPT" en la página 1044.

7. Si ha añadido el certificado de CA a un archivo de almacén de claves en el paso "4" en la página 1061, siga las instrucciones del paso "6" en la página 1063 para cifrar la contraseña del almacén de claves de CA.
8. Edite el archivo de configuración de `mqipt.conf`.
   a) Confirme que tiene la autorización adecuada para utilizar esta característica de IBM MQ Advanced estableciendo la propiedad global de **EnableAdvancedCapabilities** en `true`.
   b) Habilite el uso del almacén de claves de hardware de cifrado en la ruta estableciendo una o más de las propiedades **SSLServerKeyRingUseCryptoHardware**, **SSLServerCAKeyRingUseCryptoHardware**, **SSLServerKeyRingUseCryptoHardware**o **SSLServerKeyRingUseCryptoHardware** en `true`.

      Para obtener más información sobre las propiedades para habilitar el uso del hardware de cifrado en una ruta, consulte Propiedades de ruta de MQIPT.

      También puede utilizar hardware de cifrado con el puerto de mandatos TLS estableciendo la propiedad **SSLCommandPortKeyRingUseCryptoHardware** en `true`.
   c) Si está utilizando un archivo de almacén de claves para certificados de CA, especifique la ubicación del almacén de claves de CA estableciendo una o más de las propiedades **SSLServerCAKeyRing** o **SSLServerCAKeyRing** .

      Si configura la ruta para utilizar hardware criptográfico para el certificado de sitio y no especifica un archivo de conjunto de claves de CA, el almacén de claves de hardware criptográfico se utiliza como almacén de claves de CA.
   d) Especifique la contraseña cifrada para acceder al hardware de cifrado y al almacén de claves de CA utilizando la propiedad **SSLServerKeyRingPW**, **SSLServerCAKeyRingPW**, **SSLClientKeyRingPW**, **SSLClientCAKeyRingPW**o **SSLCommandPortKeyRingPW** .

      Establezca el valor de las propiedades **SSL*KeyRingPW** en la contraseña cifrada que ha creado el mandato **mqiptPW** .
   e) Si el hardware de cifrado contiene más de un certificado personal, especifique qué certificado MQIPT envía al servidor o cliente SSL/TLS para la autenticación.

      - Para una ruta de cliente SSL/TLS, especifique qué certificado se selecciona utilizando una o más de las propiedades **SSLClientSite*** .
      - Para una ruta de servidor SSL/TLS, especifique qué certificado se selecciona utilizando una o más de las propiedades **SSLServerSite*** .
      - Para el puerto de mandatos TLS, especifique qué certificado se selecciona utilizando la propiedad **SSLCommandPortSiteLabel** para especificar la etiqueta de certificado.

      Para obtener más información sobre cómo seleccionar certificados de un almacén de claves, consulte "Selección de certificados de un archivo de conjunto de claves en MQIPT" en la página 1044. Las propiedades para seleccionar un certificado de un almacén de claves se describen en Propiedades de ruta deMQIPT.

Por ejemplo, para utilizar un almacén de claves de hardware criptográfico para el certificado de sitio en una ruta de servidor TLS y un archivo de almacén de claves para almacenar los certificados CA para la misma ruta, añada las propiedades siguientes a la definición de ruta:

```
SSLServerKeyRingUseCryptoHardware=true
SSLServerKeyRingPW=<mqiptPW>1!gORdM4wft5d1rCgNMDEGag==!dZxhgQD2A8Ea0yeqawQvPg==
SSLServerCAKeyRing=/opt/mqipt/ssl/ca.pfx
SSLServerCAKeyRingPW=<mqiptPW>1!3Vdrpiu6kMwn0sWRCVgT5g==!LHltGLEg3OFvN8+02Re0YA==
SSLServerSiteLabel=mqiptsite
```

9. Reinicie MQIPT.

Java security manager en MQIPT

Java security manager se puede utilizar con cualquier característica de MQIPT para proporcionar un nivel de seguridad adicional.

Nota: Deprecated El uso de Java security manager con MQIPT está en desuso debido a que Java security manager ha quedado en desuso para su eliminación en un futuro release de Java.

MQIPT utiliza el Java security manager predeterminado tal como está definido en la clase `java.lang.SecurityManager`. La característica Java security manager en MQIPT puede estar habilitada o inhabilitada utilizando la propiedad global **SecurityManager**. Consulte Propiedades globales de MQIPT para obtener más información.

Java security manager utiliza dos archivos de política predeterminados:

- Un archivo de política de sistema global llamado *$MQIPT_PATH*`/java/jre/lib/security/java.policy` (donde *$MQIPT_PATH* es el directorio donde está instalado MQIPT) es utilizado por todas las instancias de una máquina virtual en un host.
- Un archivo de política específico de usuario denominado `.java.policy`, que puede existir en el directorio inicial del usuario.

También se puede utilizar un archivo de política MQIPT adicional. Deberá utilizar el archivo de política MQIPT en lugar de los archivos de política predeterminados, tal como se describe anteriormente. Consulte **SecurityManagerPolicy** en Propiedades globales deMQIPT para obtener más información.

La sintaxis del archivo de política es bastante compleja y aunque se puede cambiar utilizando un editor de texto, normalmente es más fácil utilizar el programa de utilidad Herramienta de política proporcionado con Java para realizar cambios. El programa de utilidad de la herramienta de política se puede encontrar en el directorio *$MQIPT_PATH*`/java/jre/bin` y está totalmente documentado en la documentación de Java.

Se ha proporcionado un archivo de política de ejemplo (`mqiptSample.policy`) con MQIPT para mostrarle los permisos que deben establecerse para la ejecución de MQIPT.

Debe editar el archivo de política de ejemplo para que coincida con la configuración. En concreto, tenga en cuenta que el directorio de inicio de MQIPT que contiene el archivo de configuración `mqipt.conf` podría no ser el mismo que el directorio de instalación de MQIPT, así que tenga cuidado de especificar los directorios correctos al configurar las entradas **FilePermission** en la política de seguridad.

Debe cambiar las entradas siguientes:

- La entrada **java.io.FilePermission** que otorga acceso de `lectura` y `escritura` al directorio `errors`. La vía de acceso del archivo de esta entrada debe hacer referencia al directorio de inicio de MQIPT, porque aquí es donde se encuentra el directorio `errors`. MQIPT crea archivos de captura de datos de anomalía de FFST (`AMQ*.FDC`) y archivos de rastreo (`AMQ*.TRC*`) en el directorio `errors` . Debe asegurarse de que MQIPT tiene permiso para crear archivos de rastreo y FFST en el directorio `errors`, de modo que la resolución de problemas sea posible.
- La entrada **java.io.FilePermission** que otorga acceso de `lectura` y `escritura` al directorio `logs`. La vía de acceso del archivo de esta entrada debe hacer referencia al directorio de inicio de MQIPT, porque aquí es donde se encuentra el directorio `logs`. MQIPT crea archivos de registro

de conexiones (`mqipt*.log`) en el directorio `logs`, si la propiedad global **ConnectionLog** está habilitada.

- Las entradas **java.io.FilePermission** que otorgan acceso de `lectura` y `ejecución` a los directorios del directorio de instalación de MQIPT como, por ejemplo, los directorios `bin`, `exits`, `lib` y `ssl`. Las vías de acceso de archivo de estas entradas se deben modificar para hacer referencia al directorio de instalación de MQIPT. Algunas de estas entradas se puede omitir si no son necesarias.
- Las entradas **java.net.SocketPermission** se deben modificar para controlar las conexiones en cada ruta de escucha de MQIPT. Los permiso de `escucha` y `aceptación` son necesarios para el puerto de escucha y la dirección de escucha para cada ruta de MQIPT.
- Las entradas **java.net.SocketPermission** se deben modificar para controlar las conexiones fuera de cada ruta de MQIPT. El permiso de `conexión` es necesario para los destinos de ruta, los servidores proxy o los servidores LDAP a los que se conecta la ruta MQIPT. El permiso de `resolución` es necesario cuando se especifican destinos utilizando un nombre de host en lugar de una dirección IP.

En función de la configuración, también es posible que tenga que añadir las entradas siguientes:

- Una entrada **java.io.FilePermission** para otorgar acceso de lectura al archivo de configuración `mqipt.conf`, o al directorio de inicio de MQIPT que contiene `mqipt.conf`.
- Una entrada **java.io.FilePermission** para otorgar acceso de lectura al propio archivo de política de seguridad. Esto resulta útil si una renovación de MQIPT provoca que el archivo de política de seguridad se vuelva a leer.
- Algunas entradas **java.io.FilePermission** para otorgar acceso de lectura a los archivos de conjunto de claves SSL/TLS y archivos de contraseñas de conjunto de claves. Esto solo es necesario cuando se utiliza una ruta que tiene las propiedades **SSLClient** o **SSLServer** habilitadas, o cuando se configura el puerto de mandatos TLS.
- Algunas entradas **java.io.FilePermission** para otorgar acceso de `lectura` o `ejecución` a cualquier clase de salida de MQIPT. Esto solo es necesario cuando está habilitada una salida de MQIPT. Es posible que tenga que otorgar permisos adicionales, si lo necesita la salida.

Nota: Las entradas Windows **java.io.FilePermission** deben utilizar dos caracteres de barra inclinada invertida (\\) para cada barra inclinada invertida de la vía de acceso. Esto se debe a que se utiliza una sola barra inclinada invertida como carácter de escape.

El archivo de ejemplo presupone que MQIPT se ha instalado en un sistema Windows en `C:\Program Files\IBM\MQ Internet Pass-Thru`. También presupone que el directorio inicial de MQIPT (la ubicación del archivo `mqipt.conf`) es el mismo que el directorio de instalación de MQIPT.

Si ha instalado MQIPT en otra ubicación, debe cambiar el directorio de la definición **codeBase** para hacer referencia al directorio de instalación de MQIPT. Tenga cuidado de incluir el prefijo correcto (`file:/`) y el sufijo de archivo correcto (`/lib/com.ibm.mq.ipt.jar`). En sistemas AIX and Linux , un URL de **codeBase** típico puede ser `file:/opt/mqipt/lib/com.ibm.mq.ipt.jar`, suponiendo que MQIPT esté instalado en `/opt/mqipt`.

Los permisos normalmente se definen con tres atributos. Para controlar las conexiones de socket, sus valores son:

permiso de clase
`java.net.SocketPermission`

nombre que se va a controlar
Se compone del formato `hostname:port`, donde cada componente del nombre se puede especificar mediante un comodín. El nombre de host puede ser un nombre de dominio o una dirección IP. La posición situada más a la izquierda del nombre de host puede especificarse mediante un asterisco (*). Por ejemplo, `harry.company1.com` coincidiría con cada una de estas series:

- `harry`
- `harry.company1.com`
- `*.company1.com`
- `*`

- `198.51.100.123` (que da por supuesto que esta es la dirección IP de `harry.company1.com`)

El componente de puerto del nombre se puede especificar como una dirección de puerto única o un rango de direcciones de puerto, por ejemplo:

1414
solo el puerto 1414

1414-
todas las direcciones de puerto mayores o iguales que 1414

-1414
todas las direcciones de puerto menores o iguales que 1414

1-1414
todas las direcciones de puerto entre 1 y 1414, incluidos

acción permitida
Las acciones utilizadas por `java.net.SocketPermission` son:

accept
Permitir que las conexiones se acepten desde el destino especificado

conectar
Permitir conexiones con el destino especificado

listen
Permitir que la aplicación escuche en el puerto o puertos especificados para las solicitudes de conexión

resolve
Permitir que se utilice DNS para resolver los nombres de dominio con direcciones IP

El control de Java security manager también se puede realizar a través de las propiedades del sistema `java.security.manager` y `java.security.policy` Java, pero se recomienda que utilice las propiedades **`SecurityManager`** y **`SecurityManagerPolicy`** para controlar MQIPT.

Para incluir la información de diagnóstico en los registros de rastreo y FFST. MQIPT debe acceder a determinadas propiedades del sistemas y variables de entorno de MQIPT. Siempre debe incluir las propiedades siguientes en la política de seguridad de Java:

```
permission java.util.PropertyPermission "java.home", "read";
permission java.util.PropertyPermission "java.version", "read";
permission java.util.PropertyPermission "java.runtime.version", "read";
permission java.util.PropertyPermission "java.vm.info", "read";
permission java.util.PropertyPermission "java.vm.vendor", "read";
permission java.util.PropertyPermission "os.arch", "read";
permission java.util.PropertyPermission "os.name", "read";
permission java.util.PropertyPermission "os.version", "read";
permission java.lang.RuntimePermission "getenv.MQIPT_PATH";
permission java.lang.RuntimePermission "getStackTrace";
permission javax.management.MBeanServerPermission "createMBeanServer";
permission javax.management.MBeanPermission "com.ibm.mq.ipt.IPTManager#-
[com.ibm.mq.ipt:type=IPTManager]", "registerMBean";
permission javax.management.MBeanPermission "com.ibm.mq.ipt.IPTManager#-
[com.ibm.mq.ipt:type=IPTManager]", "unregisterMBean";
permission javax.management.MBeanTrustPermission "register";
```

Si no incluye todas estas propiedades, MQIPT no funcionará correctamente y se verá afectado el diagnóstico de problemas.

Salidas de seguridad en MQIPT

Utilice una salida de seguridad para controlar el acceso a un destino, tal como se definen mediante la propiedad de ruta **`Destination`**. Se llama a la salida de seguridad en el punto cuando MQIPT recibe una solicitud de conexión de un cliente, pero antes de que realice la conexión con el destino.

Basándose en las propiedades de conexión iniciales, la salida de seguridad decide si la conexión está permitida para completarse.

Cuando se inicia una ruta, se llama a la salida de seguridad para poder inicializarse y para que esté preparada para procesar una solicitud de conexión. El proceso de inicialización se debería utilizar para cargar los datos de usuario y preparar estos datos para un acceso rápido y fácil, minimizando así el tiempo que se tarda en procesar una solicitud de conexión.

Cada ruta tiene su propia salida de seguridad.

- La propiedad **SecurityExit** se utiliza para habilitar/inhabilitar la salida de seguridad definida por el usuario.
- La propiedad **SecurityExitName** se utiliza para definir el nombre de clase de la salida de seguridad definida por el usuario.
- La propiedad **SecurityExitPath** se utiliza para definir el nombre del directorio que contiene el archivo de clase. Si esta propiedad no está establecida, se presupone que el archivo de clase se encontrará en el subdirectorio de salidas. **SecurityExitPath** también puede definir el nombre de un archivo JAR que contiene la salida de seguridad definida por el usuario.
- La propiedad **SecurityExitTimeout** es utilizada por MQIPT para determinar durante cuánto tiempo debe esperar una respuesta de la salida de seguridad al validarse una solicitud de conexión.

Consulte Propiedades de ruta de MQIPT para obtener detalles de las propiedades de salida de seguridad.

MQIPT utiliza la clase `SecurityExit` para llamar a una salida de seguridad definida por usuario. Esta clase se debe ampliar mediante la salida de seguridad definida por usuario y la mayoría de sus métodos alterados temporalmente para proporcionar la funcionalidad necesaria. Se utiliza un objeto `SecurityExitResponse` para volver a pasar los datos a MQIPT y estos datos son utilizados por MQIPT para decidir si la solicitud de conexión se debe aceptar o rechazar. El objeto `SecurityExitResponse` también puede contener un nuevo destino y una nueva dirección de puerto de destino, se utiliza para alterar temporalmente la ruta definida por las propiedades de salida de seguridad.

Se proporcionan tres salidas de seguridad de ejemplo para mostrarle cómo se puede implementar una salida de seguridad.

- `SampleSecurityExit` muestra cómo controlar el acceso a un gestor de colas IBM MQ, basándose en el nombre del canal IBM MQ. Solo permite una conexión con un nombre de canal que empieza por la serie "MQIPT." Consulte Utilización de una salida de seguridad para obtener más información.
- `SampleRoutingExit` permite el direccionamiento dinámico de solicitudes de conexión de cliente a una agrupación de servidores de IBM MQ definidos, donde cada servidor aloja un gestor de colas del mismo nombre y los mismos atributos. El ejemplo incluye un archivo de configuración que contiene una lista de nombres de servidor. Consulte Direccionamiento de solicitudes de conexión de cliente a servidores de gestor de colas de IBM MQ utilizando salidas de seguridad para obtener más información.
- `SampleOneRouteExit` permite el direccionamiento dinámico a un gestor de colas IBM MQ que se ha derivado del nombre de canal IBM MQ utilizado en la solicitud de conexión. El ejemplo incluye un archivo de configuración que contiene una correlación de nombres de gestor de colas con nombres de servidor. Consulte Direccionamiento dinámico de solicitudes de conexión de cliente para obtener más información.

Nota: MQIPT se ejecuta en una sola JVM por lo que una salida de seguridad definida por usuario podría poner en peligro el funcionamiento normal de MQIPT de una de estas formas:

- Afectar a los recursos del sistema
- Generar cuellos de botella
- Degradar el rendimiento

Debería probar los efectos de la salida de seguridad ampliamente antes de implementarla en un entorno de producción.

La clase com.ibm.mq.ipt.exit.SecurityExit en MQIPT

Esta clase y sus métodos públicos deben ser ampliados por la salida de seguridad definida por el usuario para obtener acceso a algunos datos comunes y permitir que se lleve a cabo alguna inicialización de MQIPT.

Antes de que MQIPT llame a cada método, algunas propiedades estarán disponibles para que las utilice el método. Sus valores se pueden recuperar utilizando los métodos get apropiados definidos en esta clase.

Métodos

public int init(IPTTrace)
Están disponibles las propiedades siguientes:

- puerto de escucha
- destino
- puerto de destino
- versión

MQIPT llama al método `init` cuando se inicia una ruta. Como devolución de este método, la salida de seguridad debe estar lista para validar una solicitud de conexión. Los códigos de retorno válidos son `ExitRc.OK` o `ExitRc.INIT_ERROR`.

public int refresh(IPTTrace)
Están disponibles las propiedades siguientes:

- puerto de escucha
- destino
- puerto de destino

MQIPT llama al método `refresh` cuando se renueva la configuración de MQIPT . Normalmente, esta acción se realizará cuando se haya modificado una propiedad en el archivo de configuración. MQIPT vuelve a cargar todas las propiedades del archivo de configuración para determinar qué propiedades se han modificado y si es necesario reiniciar una ruta.

Este método debe realizar una recarga de los datos externos que utilice; es decir, los datos cargados por el método **`init`**. Los códigos de retorno válidos son `ExitRc.OK` o `ExitRc.REFRESH_ERROR`.

public void close(IPTTrace)
Están disponibles las propiedades siguientes:

- puerto de escucha
- destino
- puerto de destino

MQIPT llama al método `close` cuando se detiene. Este método debe liberar los recursos del sistema que la salida ha adquirido durante su operación. MQIPT esperará a que este método se complete antes de concluir.

También se llamará a este método si anteriormente se ha habilitado una salida de seguridad, pero ahora se ha inhabilitado en el archivo de configuración.

public SecurityExitResponse validate(IPTTrace)
Están disponibles las propiedades siguientes:

- puerto de escucha
- destino
- puerto de destino
- tiempo de espera
- dirección IP de cliente
- dirección de puerto de cliente
- nombre de canal
- nombre del gestor de colas

MQIPT llama al método `validate` cuando recibe una solicitud de conexión para validar. El nombre de canal y el nombre de gestor de colas no estarán disponibles si se ha habilitado la propiedad **SSLProxyMode**, ya que esta característica solo se utiliza para crear un túnel de datos TLS y, por lo tanto, los datos que se suelen obtener del flujo inicial de datos no serán legibles.

La salida de seguridad debe devolver un objeto `SecurityExitResponse`, que contiene la información siguiente:

- código de razón (debe establecerse)
- nueva dirección de destino (opcional)
- nueva dirección de puerto de escucha de destino (opcional)
- mensaje (opcional)

El código de razón determina si la conexión es aceptada o rechazada por MQIPT. Si lo desea, los campos `newDestination` y `newDestinationPort` se pueden establecer para definir un nuevo gestor de colas de destino. Si no establece estas propiedades, se utilizarán las propiedades de ruta **Destination** y **DestinationPort** definidas en el archivo de configuración. Los mensajes devueltos se añadirán a la entrada del archivo de registro de conexión.

Se admiten los métodos siguientes para obtener los valores de las propiedades de configuración de MQIPT:

public int getListenerPort()
recupera el puerto de escucha de ruta - según lo definido por la propiedad **ListenerPort**

public String getDestination()
recupera la dirección de destino - según lo definido por la propiedad **Destination**

public int getDestinationPort()
recupera la dirección del puerto de escucha de destino - según lo definido por la propiedad **DestinationPort**

public String getClientIPAddress()
recupera la dirección IP del cliente que realiza la solicitud de conexión

public int getClientPortAddress()
recupera la dirección de puerto utilizada por el cliente que realiza la solicitud de la conexión

public int getTimeout()
recupera el valor de tiempo de espera. MQIPT esperará a que la salida de seguridad valide una solicitud - según lo definido por la propiedad **SecurityExitTimeout**

public int getConnThreadID()
recupera el ID de hebra de conexión que maneja la solicitud de conexión, que es útil para fines de depuración

public String getChannelName()
recupera el nombre de canal de IBM MQ utilizado en la solicitud de conexión

public String getQMName()
recupera el nombre de gestor de colas IBM MQ utilizado en la solicitud de conexión

public boolean getTimedout()
puede ser utilizado por la salida de seguridad para determinar si el tiempo de espera ha caducado

La clase com.ibm.mq.ipt.exit.SecurityExitResponse en MQIPT

Esta clase se utiliza para pasar una respuesta de vuelta a MQIPT desde una salida de seguridad definida por usuario y se utiliza para determinar si la solicitud de conexión se debe aceptar o rechazar.

Los objetos de este tipo solo se crean en el método de validación (consulte “La clase com.ibm.mq.ipt.exit.SecurityExit en MQIPT” en la página 1067). Hay constructores prácticos para crear estos objetos y hay métodos para cada propiedad. Consulte las salidas de seguridad para obtener más información.

La creación de un objeto SecurityExitResponse predeterminado rechaza la solicitud de conexión.

Constructores

- **public SecurityExitResponse (String dest, int destPort, int rc, String msg)**

  donde:

  – `dest` es el nuevo destino
  – `destPort` es la nueva dirección de puerto de destino
  – `rc` es el código de razón
  – `msg` es un mensaje que se añadirá a la entrada del registro de conexiones
- **public SecurityExitResponse (String dest, int destPort, int rc)**
- **public SecurityExitResponse (int rc, String msg)**
- **public SecurityExitResponse (int rc)**

Métodos

public void setDestination(String dest)
establece una nueva dirección de destino para la solicitud de conexión

public void setDestinationPort(int port) throws IPTException
define una nueva dirección de puerto de escucha de destino para la solicitud de conexión - se lanza una IPTException para una dirección de puerto no válida

public void setMessage(String msg)
añade un mensaje al registro de anotaciones de conexión

public void setReasonCode(int rc)
establece el código de razón para la solicitud de conexión.

Códigos de retorno de salida de seguridad en MQIPT

Los códigos de retorno que MQIPT reconoce al llamar a una salida de seguridad en una serie de situaciones diferentes.

Los códigos de retorno siguientes son reconocidos por MQIPT al llamar a una salida de seguridad en las situaciones siguientes:

Código de retorno	Descripción	init	Validate	Icono Renovar
ExitRc.OK	La solicitud se ha completado correctamente.	sí	sí	sí
ExitRc.INIT_ERROR	La solicitud de inicialización ha fallado, la ruta se inhabilitará.	sí		
ExitRc.REFRESH_ERROR	La solicitud de renovación ha fallado.			sí
ExitRc.NOT_AUTHORIZED	El proceso de validación ha fallado, se ha rechazado la solicitud de conexión.		sí	
ExitRc.DISABLE_SSL	La solicitud de validación ha sido satisfactoria, la conexión con el destino no utilizará SSL o TLS.		sí	

Control de número de puerto en MQIPT

Al utilizar MQIPT, es posible restringir el rango de números de puerto local que se utilizan al realizar una conexión de salida.

Establezca la propiedad **OutgoingPort** en la ruta para especificar el número de puerto local inicial y establezca **MaxConnectionThreads** para especificar el número de puertos que se van a utilizar. Por

ejemplo, si establece **OutgoingPort** en 1600 y **MaxConnectionThreads** en 20, el rango de números de puerto local para dicha ruta es 1600 - 1619.

Es responsabilidad del administrador de MQIPT asegurarse de que no hay ningún conflicto de números de puerto entre las rutas.

Si **OutgoingPort** no está definido, un valor predeterminado de 0 significa que se utiliza un número de puerto asignado por el sistema para cada conexión.

Al utilizar HTTP, el número de puertos de salida es dos veces mayor que cuando no se utiliza HTTP. En el ejemplo anterior, si la ruta utilizaba HTTP, el rango de números sería 1600 - 1639.

Consulte Asignación de números de puerto para obtener más información.

Sistemas con varios inicios

Cuando se utiliza un sistema con varios inicios, puede especificar qué dirección IP estará enlazada a una conexión de salida utilizando la propiedad **LocalAddress**. Los nombres de host no están soportados en esta propiedad.

Cifrado de contraseñas almacenadas en MQIPT

La configuración de MQIPT puede incluir contraseñas para acceder a diversos recursos, así como la contraseña para acceder a MQIPT utilizando el puerto de mandatos. Todas estas contraseñas deben protegerse cifrándose.

Acerca de esta tarea

Todas las contraseñas almacenadas para su uso por parte de MQIPT deben protegerse cifrando la contraseña con el mandato **mqiptPW** . Las contraseñas cifradas se almacenan como valores de propiedad en el archivo de configuración mqipt.conf. MQIPT puede distinguir entre contraseñas cifradas, contraseñas de texto sin formato y nombres de archivo en los valores de propiedad. Debe cifrar todas las contraseñas almacenadas para que las utilice MQIPT de este modo, ya que es el método de protección más seguro.

Para mejorar la protección de las contraseñas de conjunto de claves, vuelva a cifrar las contraseñas de conjunto de claves que se hayan cifrado anteriormente, utilizando el método de protección más reciente.

Nota: La propiedad **SSLCommandPortKeyRingPW** en el archivo de configuración mqipt.conf y la propiedad **SSLClientCAKeyRingPW** en el archivo de propiedades **mqiptAdmin** no pueden hacer referencia a los archivos de contraseñas. El mandato **mqiptPW** debe establecer los valores de estas propiedades en la salida de serie de contraseña cifrada.

Si hay un texto sin formato o una contraseña protegida débilmente en la configuración de MQIPT, se emite un mensaje de aviso cuando se inicia MQIPT o cuando se inicia una ruta.

Utilice este procedimiento para cifrar una contraseña que se almacenará para que la utilice MQIPT utilizando el método de protección más reciente.

Procedimiento

1. Opcional: Cree un archivo que contenga la clave de cifrado de contraseña, si todavía no tiene uno.

   MQIPT utiliza una clave de cifrado para cifrar las contraseñas. Puede especificar su propia clave de cifrado en un archivo. El archivo debe contener al menos un carácter y solo una línea de texto.

   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.

   Puede utilizar una clave de cifrado de contraseña distinta para cifrar las contraseñas almacenadas en el archivo de propiedades **mqiptAdmin** que la clave de cifrado utilizaba para cifrar las contraseñas en la configuración de MQIPT.

Si tiene previsto ejecutar MQIPT como servicio que se inicia automáticamente, debe crear el archivo de claves de cifrado de contraseñas con el nombre predeterminado de `mqipt_cred.key` y situarlo en el directorio de inicio de MQIPT.

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.

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.

2. Cifre la contraseña utilizando el mandato **mqiptPW**.

   La sintaxis del mandato **mqiptPW** se describe en mqiptPW (cifrar la contraseña almacenada).

   Si ha creado un archivo de claves de cifrado en el paso "1" en la página 1071, especifique el nombre de archivo utilizando el parámetro **-sf** de **mqiptPW**. Por ejemplo, se puede emitir el siguiente mandato para cifrar una contraseña utilizando la clave de cifrado en el archivo especificado por el parámetro **-sf**:

   ```
   mqiptPW -sf /opt/mqipt/mqipt_password.key
   ```

3. Especifique la contraseña que se va a cifrar cuando se le solicite.

   La contraseña cifrada será la salida de **mqiptPW**.

4. Copie la contraseña cifrada en la propiedad adecuada del archivo de configuración `mqipt.conf` o del archivo de propiedades **mqiptAdmin**.

   Por ejemplo, la línea siguiente especifica una contraseña cifrada para la contraseña de acceso de MQIPT:

   ```
   AccessPW=<mqiptPW>1!QL+2Jvj/tigKKlD7Nz8Oqw==!AMDBefOUrmPf5i1OuqV5MA==
   ```

5. Inicie MQIPT. Si ha creado un archivo de claves de cifrado de contraseña en el paso "1" en la página 1071 con un nombre que no sea el predeterminado, especifique el nombre del archivo de claves de cifrado al iniciar MQIPT.

   Puede especificar el nombre del archivo de claves de cifrado de contraseñas utilizando el parámetro **-sf** al iniciar MQIPT. Por ejemplo, emita el mandato siguiente para iniciar MQIPT utilizando la clave de cifrado en el archivo especificado por el parámetro **-sf**:

   ```
   mqipt /opt/mqipt -sf /opt/mqipt/mqipt_password.key
   ```

   Para obtener información sobre otros métodos para especificar el nombre de archivo de claves de cifrado de contraseña al iniciar MQIPT, consulte Especificación de la clave de cifrado de contraseña.

   Puede especificar el nombre del archivo de claves de cifrado de contraseña para el mandato **mqiptAdmin** utilizando la propiedad **PasswordProtectionKeyFile** en el archivo de propiedades **mqiptAdmin**.

Otras consideraciones de seguridad para MQIPT

MQIPT tiene varias funciones adicionales que ayudan a un diseñador a crear una solución segura.

- Si hay muchos clientes en una red externa que están todas intentando realizar conexiones de salida, pueden pasar todas a través de un MQIPT situado dentro del cortafuegos. El administrador del cortafuegos debe otorgar acceso externo solo al sistema MQIPT.
- MQIPT solo puede conectarse a los gestores de colas para los cuales se haya configurado explícitamente en su archivo de configuración, a menos que MQIPT esté actuando como proxy SOCKS o esté utilizando una salida de seguridad.
- MQIPT verifica que los mensajes que recibe y transmite son válidos, y que se ajustan al protocolo IBM MQ. Esto ayuda a impedir que se utilice MQIPT para los ataques de seguridad fuera del protocolo IBM MQ. Si MQIPT está actuando como un proxy SSL/TLS, cuando se han cifrado todos los protocolos y

datos de IBM MQ, MQIPT solo puede garantizar el reconocimiento inicial de SSL/TLS. En esta situación, utilice Java security manager.

- MQIPT permite que las salidas de canal ejecuten sus propios protocolos de seguridad de extremo a extremo.
- Puede restringir el número total de conexiones entrantes estableciendo la propiedad `MaxConnectionThreads`. Esto ayuda a proteger un gesto de colas interno vulnerable contra los ataques de denegación de servicio.

Archivo de configuración

Debe proteger el archivo de configuración de MQIPT, `mqipt.conf`, de ser leído por usuarios no autorizados porque podría contener información confidenciales como, por ejemplo, la contraseña **AccessPW** que controla el acceso administrativo remoto a MQIPT. Proteja todas las contraseñas especificadas en el archivo de configuración siguiendo el procedimiento de "Cifrado de contraseñas almacenadas en MQIPT" en la página 1071. Además, asegúrese de que `mqipt.conf` está protegido de modificaciones no autorizadas. Establezca los permisos de archivo del sistema operativo para `mqipt.conf`, de forma que solo la cuenta de usuario que ejecuta MQIPT pueda leer o actualizar el archivo.

Puerto de mandatos

Los puertos de mandatos de MQIPT aceptan mandatos administrativos emitidos a través de la red a una instancia remota de MQIPT mediante el mandato **mqiptAdmin**.

MQIPT se puede configurar con un puerto de mandatos que no está protegido y un puerto de mandatos que está protegido con TLS. Las conexiones al puerto de mandatos no están cifradas.

Nota: los datos enviados a través de la red, incluida la contraseña de acceso de MQIPT, pueden ser visibles para los demás usuarios de la red.

Debe considerar si necesita habilitar un puerto de mandatos y evaluar los riesgos de permitir la administración remota de MQIPT, antes de habilitar el puerto de mandatos no seguro o TLS. El mandato **mqiptAdmin** puede administrar instancias locales de MQIPT que se ejecutan con el mismo usuario que el mandato **mqiptAdmin** sin utilizar un puerto de mandatos. Por lo tanto, es posible que no tenga que habilitar un puerto de mandatos para administrar las instancias locales de MQIPT.

Si el puerto de mandatos TLS o no protegido está habilitado, debe impedir el acceso no autorizado al puerto de mandatos. Por ejemplo, debe tener en cuenta estos puntos al proteger el acceso al puerto de mandatos:

- Utilice un cortafuegos para restringir el conjunto de sistemas que se pueden conectar al puerto de mandatos de MQIPT.
- Habilite la autenticación en los puertos de mandatos utilizando las propiedades **AccessPW** y **RemoteCommandAuthentication**. Para obtener más información sobre cómo habilitar la autenticación de puerto de mandatos, consulte Autenticación de puerto de mandatos.
- Considere inhabilitar la conclusión remota con la propiedad **RemoteShutdown**.
- Considere la posibilidad de utilizar las propiedades **CommandPortListenerAddress** y **SSLCommandPortListenerAddress** para configurar los puertos de mandatos para que escuchen en una interfaz de red específica.

Para obtener más información sobre cómo utilizar el mandato **mqiptAdmin** para administrar MQIPT, consulte Administración de MQIPT utilizando la línea de mandatos.

Registros de conexión en MQIPT

MQIPT proporciona un recurso de registro de conexiones que contiene listas de todos los intentos de conexión satisfactorios y no satisfactorios.

Para cada conexión recibida o realizada por una ruta de MQIPT y para cada mandato administrativo recibido por MQIPT se graba una entrada en el registro de conexiones. El registro de conexiones

se controla utilizando las propiedades **ConnectionLog** y **MaxLogFileSize**. Consulte Propiedades globales de MQIPT para obtener más información.

Cada vez que se inicia MQIPT, se crea un nuevo registro de conexiones. En aras a la identificación, el nombre de archivo incluye la indicación de fecha y hora actual, por ejemplo:

```
mqiptYYYYMMDDHHmmSS.log
```

donde

YYYY es el año
MM es el mes
DD es el día
HH son las horas
mm son los minutos
SS son los segundos

Cuando un registro de conexiones alcanza el tamaño máximo según lo determinado por la propiedad **MaxLogFileSize**, se crea un archivo de copia de seguridad `mqipt001.log`. Se mantienen un máximo de dos archivos de copia de seguridad (`mqipt001.log` y `mqipt002.log`).

Una entrada del registro de conexiones representa cada parte de una solicitud de conexión. Una solicitud de conexión recibida por MQIPT y la nueva conexión resultante que realiza MQIPT hace que la dirección de destino aparezca como dos entradas de registro y, posteriormente, dos entradas posteriores, cuando finaliza cada conexión.

Este es el registro de conexión para una solicitud de conexión satisfactoria:

```
Wed May 15 13:13:51 BST 2013 conn accept 127.0.0.1(3842) 127.0.0.1(5000) OK 5000-0
Wed May 15 13:13:51 BST 2013 conn conn   127.0.0.1(3843) localhost(3500) OK 5000-0
Wed May 15 13:13:52 BST 2013 conn close  127.0.0.1(3842) 127.0.0.1(5000) OK 5000-0
Wed May 15 13:13:52 BST 2013 conn close  127.0.0.1(3843) localhost(3500) OK 5000-0
```

A continuación se muestra un registro de conexión para una solicitud de conexión anómala:

```
Wed May 15 14:56:40 BST 2013 conn accept 127.0.0.1(4138) 127.0.0.1(7000) OK    7000-0
Wed May 15 14:56:40 BST 2013 conn close  127.0.0.1(4138) 127.0.0.1(7000) ERROR 7000-0
 Unrecognized SSL handshake request '54'
```

Entradas de registro de conexiones

Cada entrada de registro de conexiones contiene la información siguiente:

- La hora a la que se ha creado la entrada.
- El tipo de entrada. El valor puede ser uno de los siguientes:

  admin
  Mandato administrativo

  conn
  Conexión de ruta
- El suceso que se ha producido. El valor puede ser uno de los siguientes:

  accept
  Solicitud de conexión recibida

  close
  Conexión cerrada

  conn
  Solicitud de conexión para direccionar el destino

dspipt
Muestra el mandato de MQIPT recibido

nodata
No se han recibido datos del proceso de llamada

ping
Solicitud de ping recibida

estado
Muestra el mandato de estado recibido

refr
Mandato de renovación recibido

stop
Mandato de detención recibido

- La dirección de red de origen y el número de puerto. El valor LOCAL se muestra para los mandatos administrativos emitidos localmente sin utilizar el puerto de mandatos.
- La dirección de red de destino y el número de puerto. Esto no se muestra para los mandatos administrativos emitidos localmente sin utilizar el puerto de mandatos.
- El código de terminación. El valor puede ser 0K o ERR0R.
- El identificador de hebra de MQIPT.
- Un mensaje de error opcional.

Configuración de IBM MQ Internet Pass-Thru mediante contenedores

Puede ejecutar IBM MQ Internet Pass-Thru (MQIPT) en un contenedor. La imagen base utilizada por la imagen de contenedor debe utilizar un sistema operativo Linux que esté soportado.

Procedimiento

- Está disponible una imagen de ejemplo de MQIPT Docker en el repositorio GitHub mq-container. Para crear y ejecutar el contenedor, siga las instrucciones de IBM MQ Internet Pass-Thru en Docker.

Qué hacer a continuación

Puede ver los contenedores en ejecución mediante el mandato **`docker ps`**. Para ver la salida de la consola de MQIPT que se ejecuta en un contenedor de Docker, utilice el mandato **`docker logs ${CONTAINER_ID}`**.

Configuración de colas de modalidad continua

La característica de colas de modalidad continua permite que una copia duplicada de cada mensaje colocado en una cola se entregue a una segunda cola. La configuración de las colas de modalidad continua se realiza cola por cola.

Las colas locales y de modelo tienen dos atributos nuevos relacionados con las colas de modalidad continua:

STREAMQ
Es el nombre de la cola a la que se deben entregar los mensajes en modalidad continua. Debe establecer el atributo **STREAMQ** en el nombre de otra cola.

Hay restricciones en las que las colas se pueden configurar para transmitir mensajes a otras colas, y hay restricciones en las que las colas se pueden establecer como destino para mensajes en modalidad continua. Para obtener información sobre las restricciones de transmisión de mensajes, consulte Restricciones de colas de transmisión.

STRMQOS
Es la calidad de servicio que se debe utilizar al entregar mensajes en modalidad continua.

Puede establecer el atributo **STRMQOS** en uno de dos valores:

BESTEF
Mejor esfuerzo, que es el valor predeterminado.

El gestor de colas intenta entregar una copia de cada mensaje a la cola especificada en el atributo **STREAMQ**. Si hay un problema al entregar el mensaje modalidad continua, esto no afecta a la entrega del mensaje original.

MUSTDUP
El gestor de colas intenta entregar una copia de cada mensaje a la cola de modalidad continua.

Si se produce un problema al entregar el mensaje en modalidad continua, el mensaje original no se entrega a su cola y la aplicación recibe MQCC_FAILED junto con un código de razón adecuado.

Consulte los mandatos MQSC ALTER queues, DEFINE queues y DISPLAY QUEUE y los mandatos Change, Copy y Create Queue, Inquire Queue e Inquire Queue (Response) para obtener más detalles.

Si se necesita más de una copia de cada mensaje, puede configurar el atributo **STREAMQ** para hacer referencia al nombre de una cola de alias de IBM MQ cuyo destino hace referencia a un tema de IBM MQ. Cuando se coloca un mensaje en la cola original, se publica una copia del mensaje en el tema nombrado.

Debe asegurarse de que tiene API o suscripciones administradas al objeto de tema, ya que cada suscripción recibe una copia del mensaje. El mensaje entregado a los suscriptores sigue las mismas reglas que otros mensajes de publicación/suscripción. Por ejemplo, cada mensaje tiene un nuevo identificador de mensaje y los campos de contexto de MQMD son diferentes de los del mensaje original. Para obtener más información sobre las similitudes y las diferencias entre los mensajes originales y transmitidos, consulte Mensajes en modalidad continua.

Ejemplos

Ejemplo de Mejor esfuerzo

En el ejemplo siguiente, una cola local ORDERS.QUEUE se modifica para colocar mensajes en modalidad continua en una segunda cola ANALYTICS.QUEUE. La calidad de servicio BESTEF se utiliza para asegurarse de que si hay un problema al transferir el mensaje en modalidad continua a ANALYTICS.QUEUE, por ejemplo, si ANALYTICS.QUEUE está lleno, el mensaje original todavía se puede transferir a ORDERS.QUEUE.

Este tipo de configuración se puede utilizar para realizar análisis en las órdenes que se reciben, analizando los mensajes en modalidad continua, mientras los mensajes originales se colocan en la cola de órdenes y se procesen. Una ventaja de la característica de cola de modalidad continua es que puede dejar los mensajes en modalidad continua en ANALYTICS.QUEUE, a la espera de ser procesados, sin que ello afecte a las órdenes reales que la empresa satisface.

```
DEFINE QLOCAL(ANALYTICS.QUEUE)

ALTER QLOCAL(ORDERS.QUEUE) STRMQOS(BESTEF) STREAMQ(ANALYTICS.QUEUE)
```

Nota: En el ejemplo **STRMQOS** se ha establecido en BESTEF, aunque puede dejar este atributo fuera del mandato **ALTER** , porque BESTEF es la calidad de servicio predeterminada.

Ejemplo de Debe duplicar

En este ejemplo, una cola local PAYMENTS.queue se modifica para colocar copias en modalidad continua de cada mensaje a otra cola local AUDIT.QUEUE. Es importante que cada mensaje colocado en la cola de pago se transmita a la cola de auditoría, de manera que se utilice la calidad de servicio MUSTDUP.

Si hay un problema al entregar el mensaje en modalidad continua a su cola, el mensaje original tampoco se entrega y la aplicación recibe un código de terminación y de razón adecuado. La aplicación debe reintentar la colocación de la misma forma que lo haría si solo hubiera una sola cola implicada.

```
DEFINE QLOCAL(AUDIT.QUEUE)

ALTER QLOCAL(PAYMENTS.QUEUE) STRMQOS(MUSTDUP) STREAMQ(AUDIT.QUEUE)
```

Notas:

1. No es necesario que la cola en modalidad continua exista al modificar la cola original. Sin embargo, es importante tener en cuenta que debido a que la calidad de servicio que se utiliza es MUSTDUP, los intentos de colocar mensajes en la cola original fallan hasta que se ha definido la cola de modalidad continua.
2. Cuando se utiliza un alias de cola con un destino de un objeto de tema, si no hay suscriptores, la entrega del mensaje en modalidad continua se considera satisfactoria y el mensaje original se entrega a su cola.
3. Si no se puede entregar un mensaje en modalidad continua a su cola, el gestor de colas no intenta entregarlo a la cola de mensajes no entregados. Sin embargo, si un mensaje en modalidad continua se envía a una cola remota, cuando viaja a través de un canal a otro gestor de colas, se puede entregar a una cola de mensajes no entregados siguiendo las reglas de letra muerta existentes.

Configuración de la cola de modalidad continua

No es necesario realizar ninguna configuración adicional en la cola de modalidad continua. Recibe mensajes de cualquier cola que la nombre como una cola de modalidad continua. Sin embargo, puede ser razonable tener en cuenta los valores de atributo configurados en la cola de modalidad continua.

Por ejemplo, si la cola original tiene una profundidad máxima de 100.000 y la cola de modalidad continua solo tiene una profundidad máxima de 5000, es posible que se pierdan mensajes en modalidad continua si STRMQOS se establece en BESTEF o que fallen al colocarlos si STRMQOS se establece en MUSTDUP, con un error MQRC_Q_FULL, aunque la cola original tenga mucho espacio sobrante.

Tenga en cuenta qué atributos de la cola de modalidad continua puede que deban cambiarse para que tengan los valores adecuados, en función de cómo se haya configurado la cola original.

Conceptos relacionados

Colas de modalidad continua

Avisos

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 que actualmente pueden adquirirse en su zona. Las referencias a programas, productos o servicios de IBM no pretenden establecer ni implicar que sólo puedan utilizarse dichos 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:

Intellectual Property Licensing
Legal and Intellectual Property Law
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, COMERCIABILIDAD 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 realizar mejoras y/o cambios en los productos y/o programas descritos en esta publicación en cualquier momento sin previo aviso.

Las referencias hechas en esta publicación a sitios web que no son de IBM se proporcionan sólo para la comodidad del usuario y no constituyen de modo alguno un aval de esos sitios web. Los materiales de estos sitios web no forman parte de los materiales para este producto IBM, por lo que la utilización de dichos sitios web es a 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 respeto 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, es posible que algunas mediciones se hayan estimado mediante 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. Todas las preguntas sobre las prestaciones de productos que no son 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 DERECHOS DE AUTOR:

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 IBM 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. 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 Estados Unidos y/o 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 (https://www.eclipse.org/).

Java y todas las marcas registradas y logotipos son marcas registradas de Oracle o sus afiliados.

Número Pieza:

(1P) P/N: