



# WebSphere MQ Everyplace V2.0.2



---

## Contenido

### Ejemplos de programación Java . . . . 1

Adaptadores (examples.adapters) . . . . .	1
Administración mediante la línea de mandatos (examples.administration.commandline) . . . . .	1
Administración mediante la GUI (examples.administration.console) . . . . .	1
Administración simple (examples.administration.simple) . . . . .	2
Interacción con un gestor de colas (examples.application) . . . . .	2
Seguridad (examples.attributes) . . . . .	3
Adición de una GUI de pequeño tamaño a una aplicación (examples.awt) . . . . .	4
Gestión de minicertificados (examples.certificates) . . . . .	4
Registro de sucesos (examples.eventlog) . . . . .	4

Creación y supresión de los gestores de colas (examples.install) . . . . .	5
Ampliación del puente MQ (examples.mqbridge.awt) . . . . .	5
Administración de objetos para un puente de MQ (examples.mqbridge.administration.commandline) . . . . .	6
Prueba de comunicación entre MQ y MQE (examples.mqbridge.application.GetFromMQ) . . . . .	6
Interfaz de MQE (examples.midp.mqeexampleapp) . . . . .	7
Implementación de JNI (examples.nativecode) . . . . .	7
Ejecución de un gestor de colas como cliente, servidor o servlet (examples.queuemanager) . . . . .	7
Clases de normas (examples.rules) . . . . .	8
Manejo del rastreo (examples.trace) . . . . .	8

### Índice. . . . . 9



---

## Ejemplos de programación Java

Introducción al conjunto de ejemplos en Java que se proporcionan con MQe.

En este tema se proporciona una descripción breve del conjunto de ejemplos de programación en Java que se proporcionan con MQe. Cada ejemplo muestra cómo utilizar o ampliar una función de MQe y la mayoría se describen más exhaustivamente en los temas relevantes de este centro de información.

---

### Adaptadores (examples.adapters)

Este paquete proporciona dos clases de ejemplo que cumplen la especificación de adaptadores de MQe.

#### MQeDiskFieldsAdapter

Esta clase de ejemplo funciona de forma idéntica al adaptador de campos de disco que se encuentra en `com.ibm.mqe.adapters`. Da soporte a la lectura y grabación de datos en el almacenamiento de archivos local.

#### WESAuthenticationGUIAdapter

Recorta el `WESAuthenticationAdapter` que se encuentra dentro de `com.ibm.mqe.adapters`. En este ejemplo se amplía `WESAuthenticationAdapter` mostrando mediante la visualización de un recuadro de diálogo que solicita al usuario información de inicio de sesión al conectarse a un proxy de WebSphere Everyplace.

---

### Administración mediante la línea de mandatos (examples.administration.commandline)

Este paquete contiene un conjunto de herramientas de ejemplo que sirve para crear objetos de MQe básicos desde la línea de mandatos. Cada programa es un ejemplo sencillo de cómo enviar mensajes administrativos y de cómo interpretar las respuestas.

Si utiliza estas herramientas y un script, puede configurar de forma fiable exactamente la misma configuración en varias máquinas.

---

### Administración mediante la GUI (examples.administration.console)

Este paquete contiene un conjunto de clases que implementan una interfaz gráfica de usuario sencilla (GUI) para gestionar recursos de MQe.

#### Admin

Servidor de la GUI de administración de ejemplo.

Asimismo, existe un conjunto de clases que proporciona la interfaz gráfica de usuario para cada recurso de MQe gestionado.

Se puede invocar la GUI de cualquiera de los modos siguientes:

- Mediante el archivo de proceso por lotes `ExamplesAdminConsole.bat`
- Desde la línea de mandatos:  
Ejemplo en C  

```
java examples.administration.console.Admin
```
- Desde un botón en el servidor de ejemplo `examples.awt.AwtMQeServer`.

---

## Administración simple (examples.administration.simple)

Este paquete contiene un conjunto de ejemplos que muestran cómo utilizar algunas de las funciones administrativas de MQe en sus programas. Al igual que con los ejemplos de aplicación, estos ejemplos pueden funcionar tanto con un gestor de colas local como con un gestor de colas remoto.

### Example1

Creación y supresión de una cola.

### Example2

Adición de una definición de conexión para un gestor de colas remoto.

### Example3

Consulta sobre las características de un gestor de colas y las colas de la que es propietario.

### ExampleAdminBase

Clase base de la que heredan todos los ejemplos de administración.

---

## Interacción con un gestor de colas (examples.application)

Este paquete contiene varios ejemplos que muestran las diferentes formas de interactuar con un gestor de colas. Por ejemplo, transferir un mensaje y obtener un mensaje de una cola. Todos los ejemplos pueden funcionar tanto con un gestor de colas local como con un gestor de colas remoto. Para poder ejecutar cualquiera de estas aplicaciones, en primer lugar se deben crear los gestores de colas que se utilizarán. Se puede utilizar el archivo de proceso por lotes CreateExampleQM.bat en Windows, o el script de shell CreateExampleQM en UNIX, para crear gestores de colas ExampleQM.

### Example1

Transferencia y obtención sencillas de un mensaje.

### Example2

Transferencia de varios mensajes y obtención posterior del segundo mensaje utilizando un campo de coincidencia.

### Example3

Utilización de un escucha de mensajes para detectar la llegada de nuevos mensajes.

### Example5

Bloqueo de mensajes y, posteriormente, obtención, desbloqueo y supresión de los mismos.

### Example6

Transferencia y obtención sencillas de un mensaje utilizando la entrega asegurada de mensajes.

### Example7

Transferencia y obtención sencilla de un mensaje a través de un proxy de Websphere Everyplace.

### ExampleBase

Clase base de la que heredan todos los ejemplos de aplicaciones.

Se pueden ejecutar estos ejemplos del modo siguiente:

### Windows

Mediante el archivo de proceso por lotes ExamplesMQeClientTest.bat

```
ExamplesMQeClientTest <JDK> <example no>  
  
    <remoteQMgrName> <localQMgr ini file>
```

UNIX Mediante el script de shell ExamplesMQeClientTest

```
ExamplesMQeClientTest <example no>  
  
    <remoteQMgrName> <localQMgr ini file>
```

donde

<JDK> es el nombre del entorno Java. El valor por omisión es IBM.

**Nota:** este parámetro no se utiliza en UNIX.

<example no>

es el número de ejemplo que se debe ejecutar (sufijo del nombre del ejemplo). El valor por omisión es 1 (Example1).

<remoteQMgrName>

es el nombre del gestor de colas con el que debe trabajar la aplicación. Puede ser el nombre de un gestor de colas local o remoto. Si es un gestor de colas remoto, la configuración de la conexión debe incluir una definición sobre cómo el gestor de colas local se puede comunicar con el gestor de colas remoto.

Por omisión, se utiliza el gestor de colas local, tal como está definido en el archivo ExamplesMQeClient.ini.

<localQMgrIniFile>

es un archivo ini que contiene los parámetros de inicio de un gestor de colas local. Por omisión, se utiliza ExamplesMQeClient.ini.

---

## Seguridad (examples.attributes)

Este paquete contiene varias clases que muestran cómo escribir componentes adicionales para ampliar la seguridad de MQe. Sin embargo, no están diseñadas para utilizarse en la mensajería asíncrona y no proporcionan mucha seguridad.

### NTAuthenticator

Autentificador que permite autenticar a un usuario en la base de datos de seguridad de Windows NT. Para autenticar correctamente el usuario debe tener establecidos los siguientes derechos de usuario en el sistema NT de destino:

- Actuar como parte del sistema operativo
- Inicio de sesión local
- Iniciar sesión como servicio

El autentificador NT utiliza la JNI (interfaz nativa de Java) para interactuar con la seguridad de Windows NT. El código para ello puede encontrarse en el directorio examples.nativecode. El archivo dll creado a partir de este código debe colocarse en la vía de acceso de la máquina NT que posee el recurso de destino.

### UnixAuthenticator

Autentificador que permite autenticar a un usuario mediante la contraseña UNIX o el sistema de distribución de contraseñas. El autentificador UNIX utiliza la JNI para interactuar con el sistema principal. El código para ello puede encontrarse en el directorio examples.nativecode. Si su sistema da soporte al archivo de la contraseña de sombra, deberá volver a compilar este código nativo con el distintivo establecido del preprocesador USE\_SHADOW. Asimismo, deberá asegurarse de que el código tiene privilegios suficientes para leer el archivo de distribución de contraseñas cuando se ejecuta. Este ejemplo no funciona si su sistema utiliza un servicio de conexión distribuido (como LDAP (Lightweight Directory Access Protocol)).

### LogonAuthenticator

Soporte de autenticación de inicio de sesión básico.

### UseridAuthenticator

Soporte para autenticación de *ID de usuario* básico.

Este ejemplo requiere un archivo UserIDS.txt como entrada. Este archivo debe tener el formato:

[serIDs]

User1Name=ContraseñaUsuario1

...

UserNName=ContraseñaUsuarioN

---

## Adición de una GUI de pequeño tamaño a una aplicación (examples.awt)

Este paquete proporciona un kit de herramientas para crear aplicaciones que requieren una interfaz gráfica pequeña. También contiene aplicaciones de ejemplo que proporcionan un servidor de gráficos a las funciones de MQe.

### AwtMQeServer

Un servidor de gráficos en el ejemplo `examples.queuemanager.MQeServer`. La clase `MQeTraceResourceGUI` proporciona un paquete de recursos que contiene series de caracteres traducidas para que la GUI pueda utilizarlas. `MQeTraceResourceGUI` está en el paquete `examples.trace`.

Puede utilizar el archivo de proceso por lotes `ExamplesAwtMQeServer.bat` para ejecutar esta aplicación.

### AwtMQeTrace

Un servidor de gráficos para `examples.trace.MQeTrace`.

Las clases `AwtDialog`, `AwtEvent`, `AwtFormat`, `AwtFrame` y `AwtOutputStream` proporcionan un kit de herramientas para crear aplicaciones gráficas basadas en `awt` de pequeño tamaño. Muchos de los ejemplos gráficos de MQe utilizan estas clases.

---

## Gestión de minicertificados (examples.certificates)

Este paquete contiene ejemplos para gestionar minicertificados.

### ListWTLSCertificates

En este ejemplo se utilizan los métodos de la clase `com.ibm.mqe.attributes.MQeListCertificates` para implementar un programa de línea de mandatos que liste minicertificados en un registro hasta varios niveles de detalle.

### RenewWTLSCertificates

En este ejemplo se utilizan los métodos de la clase `com.ibm.mqe.registry.MQePrivateRegistryConfigure` para implementar un programa de línea de mandatos que renueve los minicertificados de un registro. Esto se debe utilizar sólo en un registro privado.

---

## Registro de sucesos (examples.eventlog)

Este paquete contiene algunos ejemplos que muestran cómo anotar los sucesos en los diferentes recursos.

### LogToDiskFile

Sirve para anotar los sucesos en un archivo de disco.

### **LogToNTEventLog**

Escribe sucesos en el registro de sucesos de Windows NT. Esta clase utiliza JNI para interactuar con el registro de sucesos de Windows NT. El código para esto se encuentra en el directorio `examples.nativecode`.

### **LogToUnixEventLog**

Escribe los sucesos en el registro de sucesos de UNIX (que normalmente es `/var/adm/messages`). Esta clase utiliza JNI para interactuar con el sistema de registro de sucesos de UNIX. El código para ello puede encontrarse en el directorio `examples.nativecode`. El daemon del sistema de anotaciones de sucesos debería configurarse para informar de los sucesos adecuados.

---

## **Creación y supresión de los gestores de colas (examples.install)**

Este paquete contiene un conjunto de clases para crear y suprimir gestores de colas.

### **DefineQueueManager**

GUI que permite al usuario seleccionar opciones al crear un gestor de colas. Cuando se han seleccionado las opciones, este ejemplo crea un archivo ini que contiene los parámetros de inicio del gestor de colas y, a continuación, crea el gestor de colas.

### **CreateQueueManager**

Un programa de GUI que solicita el nombre y el directorio de un archivo ini que contiene los parámetros de inicio del gestor de colas. Una vez proporcionados el nombre y el directorio, se crea el gestor de colas.

### **SimpleCreateQM**

Un programa de línea de mandatos que toma como parámetro el nombre de un archivo ini que contiene los parámetros de inicio del gestor de colas. De forma alternativa, toma un parámetro que es un directorio raíz en el que se almacenan las colas. Si se encuentra un archivo ini válido, se crea un gestor de colas.

### **DeleteQueueManager**

Un programa de GUI que toma el nombre de un archivo ini que contiene los parámetros de inicio del gestor de colas. Si se encuentra un archivo ini válido, se suprimirá el gestor de colas.

### **SimplesDeleteQM**

Un programa de línea de mandatos que toma como parámetro el nombre de un archivo ini que contiene los parámetros de inicio del gestor de colas. Si se encuentra un archivo ini válido, se suprimirá el gestor de colas.

### **GetCredentials**

Un programa de GUI que toma el nombre de un archivo ini que contiene los parámetros de inicio del gestor de colas. Si se encuentra un archivo ini válido, se obtienen credenciales nuevas (certificado público y par de claves privada/pública) para el gestor de colas. El servidor de minicertificados debe estar ejecutándose y debe haberse autorizado la solicitud de un certificado nuevo para que este proceso resulte satisfactorio.

Todos los archivos de configuración utilizan los recursos y los programas de utilidad proporcionados en `ConfigResource` y `ConfigUtils`.

---

## **Ampliación del puente MQ (examples.mqbridge.awt)**

Este paquete contiene un conjunto de clases que muestran cómo utilizar y ampliar el puente de MQ. Algunos de los ejemplos amplían otros ejemplos de MQe.

### **AwtMQBridgeServer**

Esta clase es un ejemplo de una interfaz gráfica de la clase `examples.mqbridge.queuemanager.MQBridgeServer` subyacente.

El código fuente de la clase MQBridgeServer muestra cómo añadir funciones de puente al programa del servidor de MQe, siguiendo estas directrices.

Para iniciar el servidor habilitado para el puente:

1. Cree una instancia del gestor de colas de MQe base y póngala en ejecución.
2. Cree una instancia de un objeto com.ibm.mqe.mqbridge.MQeMQBridges y utilice su método activate(), pasando la misma información de archivo .ini que ha pasado al gestor de colas de MQe base.

A continuación, se podrá utilizar la función de puente.

Para detener el servidor habilitado para el puente:

1. Inhabilite la función de puente invocando el método MQeMQBridges.close(). Esto detiene todas las operaciones actuales del puente de MQ de forma controlada y concluye todas las funciones del puente de MQ.
2. Anule su referencia en el objeto MQeMQBridges y de este modo, permita que pueda desecharse.
3. Detenga y cierre el gestor de colas de MQe.

#### **ExamplesAwtMQBridgeServer.bat**

Este archivo proporciona un ejemplo de cómo invocar MQBridgeServer mediante el servidor Awt. Asimismo, muestra cómo controlar los parámetros iniciales del módulo AwtMQBridgeTrace.

#### **ExamplesAwtMQBridgeServer.ini**

Este archivo proporciona una configuración de ejemplo para un gestor de colas que soporte las funciones de puente de MQ.

---

## **Administración de objetos para un puente de MQ (examples.mqbridge.administration.commandline)**

Este paquete contiene un conjunto de herramientas de ejemplo, parecidas a las del paquete examples.administration.commandline, diseñadas para administrar los objetos necesarios para un puente de MQ.

---

## **Prueba de comunicación entre MQ y MQe (examples.mqbridge.application.GetFromMQ)**

Los programas de ejemplo que hay en este paquete son útiles para probar que MQe y MQ pueden comunicarse entre sí. Estos ejemplos son programas de enlaces de MQ que utilizan las clases Java y que se controlan mediante una sintaxis de línea de mandatos sencilla.

### **GetFromMQ**

Esta clase lee destructivamente cualquier mensaje que aparezca en una determinada cola de MQ y proporciona estadísticas de temporización cuando llega el mensaje. Opcionalmente el contenido del mensaje se puede volcar en la pantalla de salida estándar.

Este ejemplo es útil cuando se comprueba un enlace entre MQe y MQ para ver qué rendimiento se consigue entre los dos sistemas. Los scripts que se ocupan de la conectividad entre MQe y MQ pueden consultar y utilizar esta clase.

### **PutFromMQ**

Esta clase transfiere un mensaje a una cola de MQ, de modo que el usuario puede especificar la cola de destino y el gestor de colas de destino. Esta cola utiliza concretamente el formato largo del método MQQueueManager.accessQueue() para utilizar cualquier definición del alias del gestor de colas de MQe que se pueda definir en la cola de MQ.

---

## Interfaz de MQe (examples.midp.mqeexampleapp)

Este paquete contiene dos aplicaciones de ejemplo para ayudarle a comprender la interfaz de MQe. El código de ejemplo se puede dividir en tres partes:

### Servicio de mensajes (example.midp.mqeexampleapp.messageservice)

Ejecuta MQe, controla un gestor de colas y realiza funciones, como la creación de colas y el envío de mensajes. Se trata de la base de los ejemplos y permite que se puedan escribir con las mínimas invocaciones de la API de MQe. Esto también significa que para ver el código necesario con el fin de crear una cola local, por ejemplo, un usuario puede mirar simplemente la función pertinente dentro de MQeMessageService.

### Ejemplo 1: Bomba de mensajes (examples.midp.mqeexampleapp.msgpump)

Se trata de una aplicación muy sencilla que consiste en un único servidor y cliente. El cliente se establece para que envíe un mensaje al servidor cada 3 segundos y que cuando el servidor lo reciba, el usuario lo visualice. Las colas son asíncronas. Las implementaciones del cliente están disponibles para MIDP y para J2SE, mientras que el servidor sólo está disponible para J2SE.

### Ejemplo 2: Aplicación de textos (examples.midp.mqeexampleapp.textapp)

Este ejemplo es ligeramente más complejo que el primero y consiste en dos servidores y un cliente. Cuando se inicia, el cliente se tiene que registrar con el servidor de registros. El servidor de registros añade el cliente a una cola de almacenamiento y reenvío en el servidor de pasarelas y responde con un mensaje en el que se indica que la operación se ha realizado correctamente o que se ha producido una anomalía. A continuación, el cliente puede enviar mensajes definidos por el usuario al servidor de pasarelas (que él mismo visualizará). El objetivo de esta aplicación es mostrar cómo un servidor distinto se puede utilizar para crear recursos necesarios para un nuevo cliente en el sistema con el fin de beneficiar la escalabilidad de grandes redes de MQe.

---

## Implementación de JNI (examples.nativecode)

Algunos de los ejemplos exigen acceso a recursos del sistema operativo en Windows NT o UNIX (AIX y Solaris). MQe accede a dichas funciones mediante la JNI. Para Windows, el código del directorio examples\native proporciona la implementación de JNI que exigen examples.attributes.NTAuthenticator y examples.eventlog.LogToNTEventLog. Para UNIX, el código del archivo examples/native/JavaUnix.c proporciona la implementación de JNI que exigen examples.attributes.UnixAuthenticator y examples.eventlog.LogToUnixEventLog.

---

## Ejecución de un gestor de colas como cliente, servidor o servlet (examples.queuemanager)

Se puede ejecutar un gestor de colas en muchos tipos de entornos diferentes. Este paquete contiene un conjunto de ejemplos que permiten que un gestor de colas se ejecute como cliente, servidor o servlet:

### MessageWaiter

Ejemplo de cómo esperar mensajes sin utilizar el método waitFormessage que se ha dejado de utilizar.

### MQeClient

Cliente sencillo que se utiliza normalmente en un dispositivo.

### MQePrivateClient

Cliente que se puede utilizar con colas seguras y mensajería segura.

### MQeServer

Servidor que puede conectarse simultáneamente con varios gestores de colas (clientes o servidores). Suele utilizarse en una plataforma de servidor. El archivo de proceso por lotes

ExamplesAwtMQeServer.bat se puede utilizar para ejecutar el ejemplo examples.awt.AwtMQeServer que proporciona un servidor de gráficos para este servidor.

#### **MQePrivateServer**

Es parecido a MQeServer pero permite la utilización de colas seguras y de mensajería segura.

#### **MQeServlet**

Ejemplo que muestra cómo ejecutar un gestor de colas en un servlet.

#### **MQeChannelTimer**

Ejemplo que sondea el gestor de canales para que pueda establecer un tiempo de espera excedido para los canales desocupados.

#### **MQeQueueManagerUtils**

Conjunto de métodos de ayuda que configuran el inicio de varios componentes de MQe.

---

## **Clases de normas (examples.rules)**

Pueden controlarse y ampliarse las funciones base de MQe mediante normas. Algunos componentes de MQe invocan clases de normas. Estas normas proporcionan un medio de modificar las funciones del componente. Este paquete contiene las siguientes clases de normas de ejemplo:

#### **ExamplesQueueManagerRules**

Clase de normas del gestor de colas de ejemplo que periódicamente intenta transmitir los mensajes guardados.

#### **AttributeRule**

Las normas de atributos de ejemplo que controlan el uso de los atributos.

---

## **Manejo del rastreo (examples.trace)**

Este paquete contiene un manejador de rastreos de ejemplo que se puede utilizar para depurar una aplicación durante su desarrollo y para efectuar un rastreo de la aplicación una vez finalizada.

#### **MQeTrace**

Clase de rastreo de MQe base.

**AwtMQeTrace**, que está en el paquete examples.awt, proporciona un servidor de gráficos para la clase MQeTrace.

#### **MQeTraceResource**

Paquete de recursos que contiene los mensajes de rastreo que puede generar MQe.

#### **MQeTraceResourceGUI**

Esta clase contiene todo el texto traducible para los controles de la ventana de rastreo.

---

# Índice

## A

archivos  
puente, ejemplo de 5, 6

## E

ejemplo  
archivos, puente 5, 6

examples.adapters 1  
examples.administration.console 1  
examples.administration.simple 2  
examples.application 2  
examples.attributes 3  
examples.awt 4  
examples.certificates 4  
examples.eventlog 4  
examples.install 5

examples.MQ bridge, línea de  
mandatos 6  
examples.queuemanager 7  
examples.rules 8  
examples.trace 8