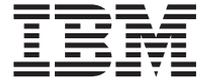


MQSeries Everyplace for Multiplatforms



# Introducción

*Versión 1.2*



MQSeries Everyplace for Multiplatforms



# Introducción

*Versión 1.2*

## Aviso

Antes de utilizar esta información y el producto al que da soporte, no olvide leer la información general del "Apéndice. Avisos" en la página 77

### Aviso sobre licencia

MQSeries Everyplace Versión 1.2 es un kit de herramientas que permite a los usuarios escribir aplicaciones MQSeries Everyplace y crear un entorno en el que ejecutarlas.

Antes de difundir este producto, o las aplicaciones que lo utilizan, en un entorno de trabajo real, asegúrese de que dispone de las licencias necesarias.

Para utilizar MQSeries Everyplace en las plataformas de servidor especificadas (para finalidades distintas al desarrollo y prueba de códigos), se deben obtener las *autorizaciones de utilización de unidad de capacidad* (que figuran en los Documentos de Titularidad y son válidas para dar soporte al uso de MQSeries Everyplace en conformidad con las tablas publicadas de grupos de precios y la unidad de capacidad) para disponer de la licencia necesaria para utilizar el programa en cada máquina y actualización de máquina.

Para utilizar el producto en plataformas de cliente especificadas (para finalidades distintas al desarrollo y prueba de códigos), se necesitan las *autorizaciones de utilización de plataforma de dispositivo* (que figuran en los Documentos de Titularidad y son válidas para dar soporte al uso de MQSeries Everyplace). Dichas licencias no autorizan al usuario a utilizar el puente de MQSeries Everyplace, ni a ejecutar las plataformas de servidor especificadas en las listas de grupos de precios de MQSeries Everyplace publicadas por IBM y que también están disponibles en la web accediendo al URL que se menciona a continuación:

Consulte <http://www.ibm.com/software/mqseries> para obtener información más detallada acerca de estas limitaciones.

### Tercera edición (mayo de 2001)

Esta edición se aplica a MQSeries Everyplace Versión 1.2 y a todos los releases y modificaciones posteriores hasta que se indique lo contrario en nuevas ediciones.

Este documento se actualiza constantemente con información nueva y mejorada. Para obtener la última edición, debe consultar la página web de la biblioteca de la familia de productos MQSeries en <http://www.ibm.com/software/mqseries/library/>.

© Copyright International Business Machines Corporation 2000, 2001. Reservados todos los derechos.

# Contenido

<b>Acerca de este manual</b> . . . . .	v
A quién va dirigido este manual. . . . .	v
Conocimientos previos necesarios . . . . .	v
Términos utilizados en este manual . . . . .	v
<b>Resumen de cambios</b> . . . . .	vii
Cambios en esta edición (GC10-3546-02) . . . . .	vii
Cambios en la edición anterior (GC10-3546-01) . . . . .	vii
<b>Capítulo 1. Visión general</b> . . . . .	1
<b>Capítulo 2. Entornos de software</b> . . . . .	3
Plataformas soportadas . . . . .	3
Entorno Java . . . . .	4
Requisitos de almacenamiento . . . . .	4
<b>Capítulo 3. La familia MQSeries</b> . . . . .	5
Productos distribuidos y de sistema principal MQSeries. . . . .	6
MQSeries Everyplace . . . . .	7
<b>Capítulo 4. Requisitos del producto</b> . . . . .	11
Posibilidades . . . . .	11
Aplicaciones. . . . .	11
Requisitos de los usuarios . . . . .	12
<b>Capítulo 5. Conceptos del producto</b> . . . . .	15
Introducción. . . . .	15
Objetos de mensaje . . . . .	16
Formato de los datos de las operaciones de vuelco . . . . .	21
Colas . . . . .	21
Gestores de colas. . . . .	27
Configuración del gestor de colas . . . . .	31
Operaciones de los gestores de colas . . . . .	34
Conexiones . . . . .	35
Administración . . . . .	37
Mensajes de administración . . . . .	37
Administración selectiva. . . . .	39
Supervisión y otras acciones relacionadas. . . . .	39
Canales dinámicos . . . . .	40
Adaptadores . . . . .	41
Gestión de conexiones de marcación . . . . .	41
Rastreo . . . . .	42
Anotaciones de sucesos . . . . .	42
Entrega de mensajes. . . . .	42
Entrega de mensajes asíncrona . . . . .	42
Entrega de mensajes síncrona . . . . .	43
Seguridad . . . . .	43

Seguridad local en MQSeries Everyplace . . . . .	45
Seguridad MQSeries Everyplace basada en colas . . . . .	45
Seguridad a nivel de mensajes . . . . .	46
El registro . . . . .	48
Entidades autentificables de MQSeries Everyplace . . . . .	48
El registro privado y las credenciales . . . . .	48
Registro automático . . . . .	49
Registro público y reproducción de certificados . . . . .	49
Utilización de los servicios de registro por parte de las aplicaciones . . . . .	49
Servicio de emisión de minicertificado por omisión . . . . .	50
La interfaz de seguridad. . . . .	50
Personalización . . . . .	51
Normas . . . . .	51
Tipos de conexiones . . . . .	52
Conexión de igual a igual . . . . .	53
Conexión cliente-servidor . . . . .	53
Tipos de conexiones diferentes . . . . .	54
Clases . . . . .	54
Carga de las aplicaciones . . . . .	55
<b>Capítulo 6. Redes MQSeries Everyplace y MQSeries . . . . .</b>	<b>57</b>
Interfaz con MQSeries . . . . .	57
Conversión de mensajes . . . . .	64
Función . . . . .	65
Compatibilidad . . . . .	66
Entrega asegurada . . . . .	67
<b>Capítulo 7. Interfaces de programación . . . . .</b>	<b>69</b>
<b>Capítulo 8. Cómo empezar con MQSeries Everyplace . . . . .</b>	<b>71</b>
Utilización de MQSeries Everyplace . . . . .	72
Cómo adquirir experiencia . . . . .	73
Primera utilización de ES02: MQE_Explorer . . . . .	73
<b>Apéndice. Avisos . . . . .</b>	<b>77</b>
Marcas registradas . . . . .	78
<b>Glosario . . . . .</b>	<b>79</b>
<b>Bibliografía. . . . .</b>	<b>83</b>
<b>Índice. . . . .</b>	<b>85</b>
<b>Envío de comentarios a IBM. . . . .</b>	<b>87</b>

---

## Acerca de este manual

Este manual es una introducción general a MQSeries Everyplace for Multiplatforms (aunque en el libro se le denomine generalmente MQSeries Everyplace). Trata sobre los conceptos del producto y su relación con otros productos MQSeries.

Para obtener información más detallada sobre la API de MQSeries Everyplace y sobre cómo utilizarla para crear aplicaciones MQSeries Everyplace, consulte la publicación *MQSeries Everyplace for Multiplatforms Programming Reference* y la publicación *MQSeries Everyplace for Multiplatforms Guía de programación*.

Si desea obtener información relacionada con la utilización de otros lenguajes de programación con MQSeries Everyplace for Multiplatforms, consulte la publicación *MQSeries Everyplace for Multiplatforms Native Client Information*

Si desea información sobre los procedimientos de instalación de MQSeries Everyplace for Multiplatforms consulte *MQSeries Everyplace for Multiplatforms Léame primero*

Este documento se actualiza constantemente con información nueva y mejorada. Para obtener la última edición, debe consultar la página web de la biblioteca de la familia de productos MQSeries en <http://www.ibm.com/software/ts/mqseries/library/>.

---

## A quién va dirigido este manual

Este manual va dirigido a todo aquel que esté interesado en utilizar un sistema de mensajería asegurada en dispositivos ligeros como, por ejemplo, sensores, teléfonos, PDA (Personal Digital Assistants) y portátiles, y a quienes necesiten ampliar el ámbito de una red de mensajería MQSeries Everyplace.

---

## Conocimientos previos necesarios

No es necesario tener conocimientos previos para leer esta información, pero es mejor conocer de antemano los conceptos básicos de la mensajería asegurada.

Si no tiene estos conocimientos iniciales, es posible que la lectura del manual siguiente de MQSeries le resulte útil:

- *MQSeries An Introduction to Messaging and Queuing*

Este manual está disponible en copia software en la sección Book de la biblioteca MQSeries en línea. Se puede consultar desde el sitio web de MQSeries, en el URL: <http://www.ibm.com/software/ts/MQSeries/library/>

---

## Términos utilizados en este manual

En esta publicación se utilizan los términos siguientes:

### **Familia MQSeries**

hace referencia a los siguientes productos MQSeries:

- **MQSeries Workflow** simplifica la integración en toda la empresa ya que automatiza los procesos en los que participan los empleados y las aplicaciones
- **MQSeries Integrator** es un software potente de intermediario de mensajes que proporciona un direccionamiento inteligente de mensajes en tiempo real, basado en normas, que además transforma y da formato al contenido
- **MQSeries Mensajería** permite las conexiones entre diferentes sistemas, desde sistemas de sobremesa a sistemas principales, con funciones de mensajería de calidad para la empresa y soporte para más de 35 plataformas

### **Mensajería MQSeries**

hace referencia a los grupos de productos de mensajería siguientes:

- **Mensajería distribuida:** MQSeries para Windows NT, AIX, AS/400, HP-UX, Sun Solaris y otras plataformas
- **Mensajería de sistema principal:** MQSeries para OS/390
- **Mensajería general:** MQSeries Everyplace

### **MQSeries**

hace referencia a los siguientes tres grupos de productos de mensajería MQSeries:

- Mensajería distribuida
- Mensajería de sistema principal
- Mensajería de estación de trabajo

### **MQSeries Everyplace**

Hace referencia al tercer grupo de productos de mensajería MQSeries, mensajería general.

### **Plataforma de dispositivo**

Sistema pequeño que puede ejecutar MQSeries Everyplace sólo como un cliente.

### **Plataforma de servidor**

Sistema de cualquier tamaño que puede ejecutar MQSeries Everyplace como un servidor o un cliente.

### **Pasarela**

Sistema de cualquier tamaño que ejecuta programas MQSeries Everyplace que incluyen la función de puente de MQSeries.

---

## Resumen de cambios

Esta sección describe los cambios en esta edición de *MQSeries Everyplace for Multiplatforms Introducción*. Los cambios efectuados desde la última edición del libro están marcados mediante líneas verticales a la izquierda de los cambios.

---

### Cambios en esta edición (GC10-3546-02)

---

### Cambios en la edición anterior (GC10-3546-01)

Se ha añadido la información siguiente:

- Información para el uso de MQSeries Everyplace en AIX y Solaris.
- Requerimientos de almacenamiento
- Hoja de comentarios



---

## Capítulo 1. Visión general

MQSeries Everyplace pertenece a la familia MQSeries de productos comerciales de mensajería. Se ha diseñado para satisfacer las necesidades de mensajería de dispositivos ligeros como, por ejemplo, sensores, teléfonos, PDA (Personal Digital Assistants) y ordenadores portátiles, así como para ofrecer soporte para la movilidad necesaria y los requisitos que surgen cuando se utilizan redes de comunicaciones frágiles. Mantiene la calidad de servicio estándar en MQSeries, proporcionando la función de entrega asegurada de una sola vez, e intercambia mensajes con otros miembros de la familia. Dado que muchas aplicaciones MQSeries Everyplace se ejecutan fuera del entorno de protección de un cortafuegos de Internet, también presenta métodos de seguridad sofisticados.

Los dispositivos ligeros necesitan que el subsistema de mensajería utilice de forma frugal los recursos del sistema y, debido a esto, MQSeries Everyplace ofrece interfaces y funciones adaptadas adecuadas para el conjunto de clientes, sin pretender facilitar exactamente las mismas posibilidades que los demás miembros de la familia. Por otra parte, no incluye una función exclusiva para ofrecer soporte para sus clases de usuario determinadas como, por ejemplo, amplias disposiciones de seguridad, objetos de mensaje, mensajería síncrona y asíncrona, acceso de colas remotas y transferencia y extracción de mensajes.

MQSeries Everyplace también se ha diseñado para que pueda integrarse correctamente con otros miembros de la familia de sistemas de IBM y con otros componentes de WebSphere Application Server.



---

## Capítulo 2. Entornos de software

---

### Plataformas soportadas

MQSeries Everyplace sólo se puede instalar directamente en plataformas de servidor determinadas. Para transferir programas y clases Java a otras plataformas, se debe utilizar un programa adecuado para bajar o transferir archivos (que no se proporciona).

#### Plataformas soportadas directamente con el soporte de instalación

A continuación se indican las plataformas en las que se puede instalar el producto utilizando las herramientas incorporadas.

- Windows NT v4
- Windows 2000
- Windows 95/98/ME
- AIX Versión 4.3
- Sun Solaris Versión 7 u 8
- Linux Intel Kernel 2.2 (instalado utilizando un archivo zip)
- HP-UX 11.0 (instalado utilizando un archivo zip)

#### Plataformas soportadas directamente sin el soporte de instalación

Se da soporte a las plataformas siguientes para la prueba y difusión de MQSeries Everyplace, pero la instalación sólo se puede realizar mediante la transferencia de archivos de otra plataforma.

- WinCE 2.1 que se ejecute en dispositivos HP Jornada (Modelos 680 u 820)
- EPOC 32 bits Release 5 que se ejecute en dispositivos Psion (5MX Pro o NetBook)
- PalmOS, V3.0 o posterior que se ejecute en Palm V e IBM Workpad C3
- IBM 4690 OS con Java

#### Plataformas soportadas indirectamente

Se pueden utilizar las plataformas siguientes, pero sólo se ofrece soporte para las mismas si su entorno Java es totalmente compatible con el de las plataformas soportadas directamente. Los problemas sólo se pueden determinar en una de las plataformas probadas enumeradas anteriormente.

- Linux en zSeries en el que se ejecute el Kernel 2.2
- iSeries
- OS/2
- EPOC (en dispositivos que no sean los de la lista anterior)
- WinCE (en dispositivos que no sean los de la lista anterior)
- QNX Neutrino
- Pocket PC
- PalmOS (en dispositivos que no sean los de la lista anterior)
- Cualquier otra plataforma en la que se ejecute uno de los entornos Java enumerados en el apartado "Entorno Java" en la página 4

---

### Entorno Java

Se necesita uno de los entornos de ejecución Java siguientes:

- Entorno de ejecución IBM Java (JVM 1.3 o posterior), incluido Java Micro Edition
- Cualquier entorno Java que tenga la certificación Java de Sun (V1.1 o posterior) <sup>1</sup>

**Nota:** Java necesita ser totalmente compatible con lo que se ha probado en una de las plataformas siguientes para que el servicio esté disponible

- Dispositivos HP Jornada (Modelos 680 u 820) en los que se ejecute el sistema operativo Windows CE
- Dispositivos Psion (5MX Pro o NetBook) en los que se ejecute el sistema operativo EPOC
- Una de las plataformas de servidor de las listas anteriores de plataformas soportadas directamente

Para la operación del puente de MQSeries se necesita MQSeries Classes for Java. Debe comprobar el nivel de Java necesario para ejecutar la versión de MQSeries Classes for Java.

---

### Requisitos de almacenamiento

En la tabla siguiente se muestra el almacenamiento que necesita para llevar a cabo la instalación de MQSeries Everyplace.

*Tabla 1. Almacenamiento necesario para realizar la instalación*

Sistema operativo	Almacenamiento necesario
Windows NT (sistema de archivos = NTFS)	26 Mb
AIX	29 Mb
Solaris	27 Mb

En la tabla siguiente se muestra el almacenamiento que necesita para los archivos de MQSeries Everyplace después de la instalación.

*Tabla 2. Almacenamiento necesario para MQSeries Everyplace*

Sistema operativo	Almacenamiento necesario
Windows NT (sistema de archivos = NTFS)	9,5 Mb
AIX	11 Mb
Solaris	10 Mb

---

1. Pueden producirse problemas si ejecuta el instalador bajo JVM de Sun con el compilador JIT (Just In Time) habilitado. Si utiliza JVM de Sun se aconseja inhabilitar el compilador JIT utilizando el mandato: `java -Djava.compiler=NONE install`

## Capítulo 3. La familia MQSeries

La familia MQSeries incluye muchos productos que cubren una gran gama de posibilidades como se muestra en la Figura 1

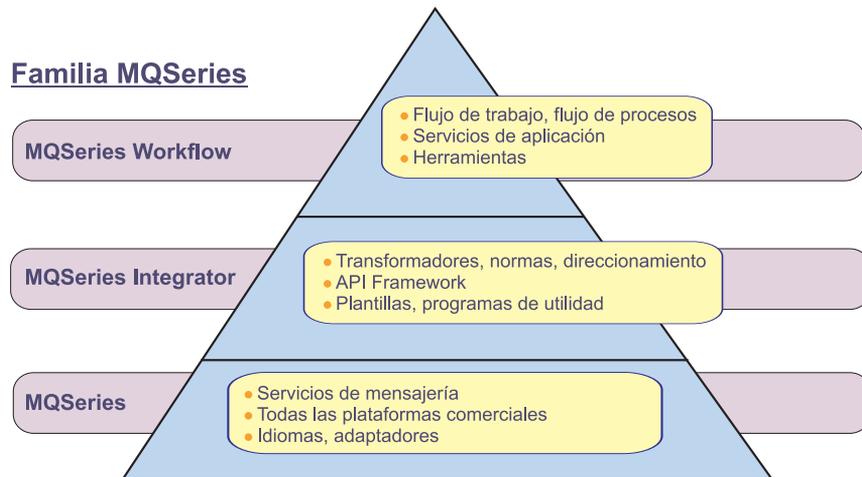


Figura 1. La familia MQSeries

- **MQSeries Workflow** simplifica la integración en toda la empresa ya que automatiza los procesos en los que participan los empleados y las aplicaciones
- **MQSeries Integrator** es un software potente de intermediario de mensajes que proporciona un direccionamiento inteligente de mensajes en tiempo real, basado en normas, que además transforma y da formato al contenido
- **MQSeries Mensajería** permite interconectar diferentes sistemas, desde sistemas de sobremesa a sistemas principales, con funciones de mensajería de calidad para la empresa y soporte para más de 35 plataformas

Tanto los productos MQSeries Workflow como MQSeries Integrator se benefician de las posibilidades de conexión que proporciona la capa de mensajería de MQSeries.

Las funciones de mensajería de la familia MQSeries las proporcionan los productos MQSeries y MQSeries Everyplace, ya que cada uno de ellos está diseñado para dar soporte a una o más plataformas de servidor de hardware y a los sistemas operativos asociados. Dada la extensa variedad de soporte de plataformas, estos productos individuales se han organizado en grupos de productos, según su función y diseño comunes. Existen tres grupos de productos de este tipo:

- **Mensajería distribuida:** MQSeries para Windows NT, AIX, AS/400, HP-UX, Sun Solaris y otras plataformas
- **Mensajería de sistema principal:** MQSeries para OS/390
- **Mensajería general:** MQSeries Everyplace

## Familia MQSeries

La mensajería por sí sola, independientemente del producto o grupo de productos determinado, está basada en los gestores de colas. Los gestores de colas gestionan las colas que pueden almacenar mensajes. Las aplicaciones se comunican con un gestor de colas locales y obtienen o transfieren los mensajes a las colas. Si se transfiere un mensaje a una cola remota, esto es, una cola de la que es propietario un gestor de colas remotas, el mensaje se transmite a través de los canales al gestor de colas remotas. De este modo, los mensajes pueden pasar por uno o más gestores de colas intermediarios antes de llegar a su destino. La esencia de la función de mensajería radica en separar la aplicación emisora de la aplicación receptora, poniendo los mensajes en cola durante el trayecto si es necesario. Todos los productos MQSeries para mensajería están relacionados con los mismos elementos básicos, esto es, con gestores de colas, colas, mensajes y canales, aunque en el fondo existen muchas diferencias.

### Productos distribuidos y de sistema principal MQSeries

Los productos de mensajería distribuida y de sistema principal MQSeries se utilizan para dar soporte a muchas configuraciones de red diferentes, y en todas ellas participan clientes y servidores.<sup>2</sup> En la Figura 2 se ilustran algunos ejemplos.

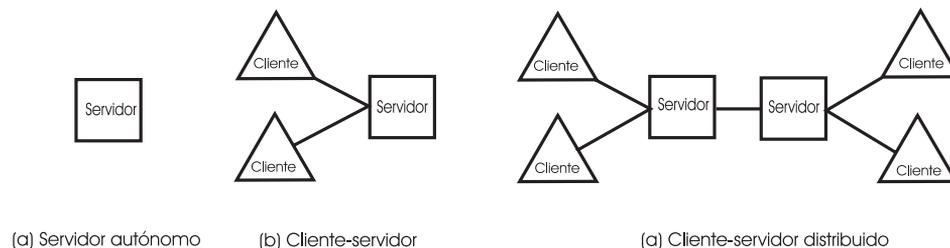


Figura 2. Configuraciones sencillas de sistema principal y distribuida

En el caso más sencillo se configura un servidor autónomo que ejecuta un gestor de colas. En este servidor se ejecutan una o varias aplicaciones que intercambian mensajes a través de las colas. Una configuración alternativa es la de cliente-servidor. En este caso, el gestor de colas sólo existe en el servidor, pero todos los clientes tienen acceso a éste a través de un *canal de cliente*. El canal de cliente es un enlace de comunicaciones bidireccional en el que fluye un tipo de protocolo exclusivo de MQSeries en el que se implementa algo similar a una llamada RPC, esto es, una llamada de procedimiento remoto. Las aplicaciones se pueden ejecutar en los clientes y acceder a las colas del servidor. Una de las ventajas de la configuración cliente-servidor es que la infraestructura de la función de mensajería del cliente es sencilla y depende del gestor de colas del servidor. Una desventaja es que los clientes y el servidor asociado a ellos operan de forma síncrona y, por lo tanto, requieren que el canal del cliente esté siempre disponible.

2. Tenga en cuenta que estos términos tienen significados muy específicos dentro de los productos de mensajería de estación de trabajo, distribuidos y de sistema principal MQSeries que no siempre se corresponden con el uso que se les da comúnmente.

La configuración cliente-servidor distribuida representa un caso más complejo en el que participan varios servidores. En este tipo de configuraciones, los servidores intercambian mensajes a través de *canales de mensajes*. Los canales de mensajes son unidireccionales y poseen un protocolo que se ha diseñado para proporcionar un intercambio seguro y asíncrono de los datos de mensajes. Estos canales de mensajes deben estar disponibles para que los clientes puedan continuar el proceso, aunque no puede haber un flujo de mensajes entre los servidores cuando las comunicaciones no están disponibles.

---

## MQSeries Everyplace

MQSeries Everyplace ofrece soporte para una gran variedad de configuraciones de red al proporcionar gestores de colas, teniendo habilitadas cada uno de ellos las posibilidades adecuadas. No existe ningún concepto de cliente o servidor como en los productos distribuidos o de sistema principal MQSeries. Los gestores de colas MQSeries Everyplace pueden actuar como clientes o servidores tradicionales pero, de hecho, cada uno de ellos es, simplemente, un gestor de colas habilitado para realizar tareas definidas por la aplicación. Por ejemplo, un gestor de colas MQSeries Everyplace se puede configurar con o sin colas locales. Con colas locales puede almacenar mensajes localmente y, por tanto, ofrecer aplicaciones de mensajería asíncrona, pero sin colas locales se restringe a la mensajería síncrona. Otro ejemplo de la configuración adaptada es que un gestor de colas se puede configurar con o sin posibilidades de puente. Con el puente dispone de la posibilidad de intercambiar mensajes con gestores de colas de estación de trabajo o de sistema principal MQSeries. En cambio, sin el puente sólo se puede comunicar directamente con otros gestores de colas MQSeries Everyplace (aunque puede comunicarse de modo indirecto a través de otros gestores de colas de la red que tengan posibilidades de puente).

**Nota:** Un nuevo nodo para MQSeries Integrator (MQSI) permite la conexión de MQSeries Everyplace directamente, sin utilizar el puente de MQSeries.

Los gestores de colas MQSeries Everyplace utilizan *canales dinámicos* para intercambiar información. Estos canales dinámicos tienen propiedades diferentes de los *canales de cliente* y de los *canales de mensajes* que utilizan otros miembros de la familia. Algunas de las características más importantes de los canales dinámicos son las siguientes:

- Soporte de mensajería *síncrona* y *asíncrona*: La mensajería síncrona proporciona un servicio de transmisión directamente de la aplicación de origen a la aplicación de destino, sin la colocación en colas en el gestor de colas de origen. La mensajería asíncrona es un servicio de transmisión del gestor de colas de origen a la cola de destino, con la posibilidad de la colocación en colas en el gestor de colas de origen.
- Soporte de *servicio global*: Los canales van del gestor de colas de origen a un gestor de colas de destino, pudiéndose ejecutar a través de gestores de colas intermediarios. El protocolo de transporte subyacente utilizado puede cambiar a medida que el canal pasa a través de dichos intermediarios.
- Soporte para *compresión*, *cifrado* y *autenticación*: Los canales disponen de estas características de seguridad para proteger los datos que están transportando.
- Soporte para operación *de igual a igual* y *cliente-servidor*: Los canales definidos como de igual a igual son funcionalmente simétricos, de modo que tanto el origen

## Familia MQSeries

como el destino pueden iniciar una operación a través del canal. Los canales cliente-servidor son de petición/respuesta. El cliente efectúa una petición al servidor y el servidor responde a la misma. (Tenga en cuenta que esto no restringe el flujo de mensajes. Los mensajes pueden fluir del cliente al servidor y del servidor al cliente).

La opción de configuración de puente permite que un gestor de colas MQSeries Everyplace se comunique con gestores de colas distribuidos o de sistema principal MQSeries a través de *canales de cliente*. El componente de puente gestiona una agrupación de canales de cliente que pueden conectarse a uno o más gestores de colas distribuidos o de sistema principal. En una única red se pueden configurar varios gestores de colas MQSeries Everyplace habilitados para puente para ofrecer la fiabilidad, el rendimiento y las posibilidades necesarias.

En los diagramas siguientes se muestran algunas configuraciones MQSeries Everyplace habituales, pero arbitrarias. Con el fin de simplificar su comprensión, en los diagramas sólo se muestran las conexiones directas que se han definido. También se pueden definir conexiones indirectas que aprovechen las conexiones directas. Para representar un canal de igual a igual se utiliza una línea con dos puntas de flecha, y para representar un canal cliente-servidor se utiliza una flecha que señala al servidor. Los clientes pueden utilizar el canal cliente-servidor para enviar mensajes al servidor y para extraer los mensajes destinados a ellos desde dicho servidor. Las líneas que no tienen ninguna flecha indican los canales de cliente MQSeries que permiten las comunicaciones entre MQSeries Everyplace y MQSeries.



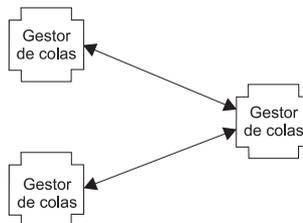
(a) Gestor de colas autónomo

(a) Muestra un gestor de colas autónomo que se utiliza para ofrecer soporte para una o más aplicaciones que usan colas para intercambiar datos.



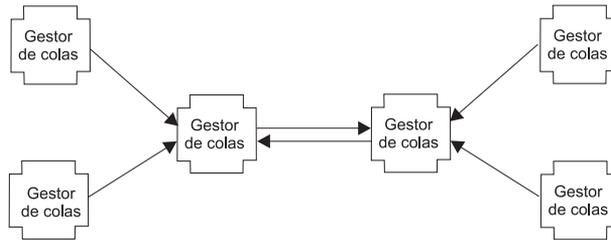
(b) Configuración de igual a igual

(b) Muestra dos gestores de colas interconectados a través de un canal de igual a igual.



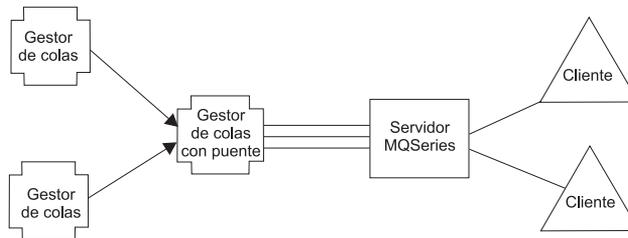
(b) Configuración de igual a igual

(c) Muestra un segundo canal de igual a igual directo a un tercer gestor de colas. Los tres pueden utilizar las conexiones directas subyacentes para intercambiar datos entre sí.



(d) Una red pequeña

(d) Muestra una configuración de red pequeña en la que los gestores de colas del servidor central utilizan un par de canales cliente-servidor directos para intercambiar información. Cada uno de los gestores de colas de cliente utiliza un canal cliente-servidor directo para conectar con uno de los gestores de colas del servidor.



(E) Una red de la familia MQSeries integrada

(e) Muestra una configuración de MQSeries Everyplace en la que uno de los gestores de colas se ha configurado con la opción de puente y la agrupación de todos los canales de cliente se ha dirigido a un único servidor distribuido/sistema principal MQSeries de destino.



---

## Capítulo 4. Requisitos del producto

Este capítulo describe los requisitos que han marcado el diseño y la implementación de MQSeries Everyplace.

---

### Posibilidades

MQSeries Everyplace amplía el ámbito de las funciones de mensajería de la familia MQSeries de esta forma:

- Da soporte a dispositivos de gama inferior como por ejemplo, PDA, teléfonos y sensores, y de este modo les permite participar en una red de mensajería de MQSeries. MQSeries Everyplace también da soporte a dispositivos intermedios como, por ejemplo, ordenadores portátiles, estaciones de trabajo y plataformas de sistema principal y distribuidas. MQSeries Everyplace ofrece la misma calidad de servicio en la entrega asegurada de mensajes de una sola vez, y permite el intercambio de mensajes con otros miembros de la familia.
- Ofrece recursos de mensajería ligeros.
- Proporciona una amplia gama de funciones de seguridad que protegen los mensajes, las colas y los datos asociados, tanto durante su almacenamiento como en la transmisión.
- Funciona de forma eficaz en entornos de comunicaciones hostiles en los que las redes son inestables o en los que el ancho de banda presenta muchas limitaciones. MQSeries Everyplace presenta un protocolo de transmisión eficaz y funciones de recuperación automática ante anomalías en los enlaces de comunicaciones.
- Da soporte a usuarios con movilidad geográfica ya que permite que los puntos de conectividad de la red varíen al mismo tiempo que lo hacen los dispositivos itinerantes. MQSeries Everyplace también permite controlar el comportamiento de éstos en aquellas condiciones en las que recursos como la batería o las redes presentan limitaciones o no funcionan correctamente.
- Realiza las operaciones a través de cortafuegos configurados adecuadamente.
- Reduce al mínimo las tareas de administración del usuario, de tal modo que la presencia de MQSeries Everyplace en un dispositivo prácticamente se puede ocultar. Esto convierte a MQSeries Everyplace en una base adecuada para crear aplicaciones prácticas.
- Se puede personalizar y ampliar fácilmente, utilizando las normas suministradas por las aplicaciones y de otras clases que modifican el comportamiento, o mediante la creación de subclases de las clases de objetos base para, por ejemplo, representar diferentes tipos de mensajes.

---

### Aplicaciones

Las aplicaciones posibles de MQSeries Everyplace son muchas y variadas, pero se supone que un gran número de éstas serán aplicaciones personalizadas desarrolladas para grupos de usuarios determinados. En la lista siguiente se ofrecen algunos ejemplos:

## requisitos del producto

- **Aplicaciones de venta al por menor:** para la suministro de pequeñas cantidades de transacciones de caja a sistemas principales como, por ejemplo, intermediarios de mensajes
- **Aplicaciones de consumidores:** para realizar compras en un supermercado desde el hogar a través de un PDA, para acumular datos sobre las preferencias de los pasajeros de las líneas aéreas, para las transacciones financieras desde un teléfono móvil
- **Aplicaciones de control:** para reunir e integrar los datos procedentes los sensores de los oleoductos y transmitidos vía satélite, para operar de forma remota equipos (como, por ejemplo, válvulas) con un nivel de seguridad que garantice la validez de la operación
- **Para el personal con movilidad geográfica:** para los representantes profesionales (de agencias de seguros), para editar rápidamente comprobantes de recibo del cliente en empresas de paquetería, para que los camareros de los restaurantes de comida rápida se comuniquen con la cocina, para los marcadores de los torneos de golf, para sistemas móviles seguros, para sistemas de mensajería de la policía, para información laboral para trabajadores de empresas de servicio público en situaciones en las que suelen perderse las comunicaciones, para la lectura de contadores domésticos.
- **Productividad personal:** para reproducción de agendas/calendarios, para reproducción de bases de datos, para reducción del tamaño de los portátiles.

---

## Requisitos de los usuarios

El diseño de MQSeries Everyplace se ha visto influido de este modo por la demanda de los usuarios:

- **Administración:** su configuración y mantenimiento son mínimos, da soporte a funciones de administración locales y remotas, permite ampliar y personalizar las funciones de administración para que se ajusten a las necesidades de determinadas aplicaciones, acentúa la detección y recuperación automáticas y proporciona elementos de administración independientes que se pueden utilizar de forma selectiva.
- **Comunicaciones:** incorpora un protocolo de transmisión muy eficaz, el número cabeceras es mínimo y los mensajes no contienen campos obligatorios (con la excepción de un identificador exclusivo), permite cambiar la codificación de los datos, da soporte a la compresión, cifrado y autenticación, permite negociar entre los dos extremos de la comunicación las características de compresión y seguridad, presenta la posibilidad de cruzar fácilmente los cortafuegos y la opción de adaptadores de comunicaciones conectables.
- **Compatibilidad:** incorpora la calidad de servicio de MQSeries y el intercambio uniforme de mensajes, permite la comunicación con los sistemas MQSeries existentes sin realizar modificaciones en las aplicaciones y un control flexible del intercambio de mensajes entre MQSeries y MQSeries Everyplace .
- **Espacio:** para el dispositivo Palm, 64 Kbytes. Para dispositivos Java, 100 Kbytes de espacio mínimo de archivos de clase.

- **Función:** permite la mensajería síncrona y asíncrona, acceder a mensajes almacenados en las colas locales o remotas, utilizar cualquier campo del mensaje para realizar una recuperación selectiva y realizar un control selectivo del soporte de copias de seguridad de una cola.
- **Soporte de normas:** permite controlar muchos de los aspectos del comportamiento de las funciones mediante normas que regirán, por ejemplo, cuándo se han de enviar los mensajes, la frecuencia de reintentos de un enlace de comunicaciones, qué hacer cuando un mensaje es demasiado grande o cuando una cola de destino está llena.
- **Seguridad:** presenta un soporte completo de las funciones de seguridad y autenticación y orden de no rechazo, incorpora funciones de seguridad tanto a nivel de mensajes como a nivel de colas, protege el sistema de mensajería contra violaciones de seguridad, permite utilizar los algoritmos estándar del sector para obtener funciones de seguridad adicionales, se puede integrar con credenciales de usuario del sistema operativo, permite atenerse a los requisitos de seguridad nacionales y presenta un soporte de seguridad que puede modificarse a medida que los mensajes cruzan fronteras.

## requisitos del producto

# Capítulo 5. Conceptos del producto

## Introducción

Los elementos fundamentales del modelo de programación de MQSeries Everyplace son los mensajes, las colas y los gestores de colas. Los mensajes de MQSeries Everyplace son objetos cuyo contenido definen las aplicaciones. Estos mensajes, cuando se almacenan, se conservan en una cola, y pueden desplazarse a través de una red MQSeries Everyplace. Los mensajes pueden dirigirse a una cola de destino, si se especifica el nombre de la cola y el nombre del gestor de colas de destino. Las aplicaciones colocan los mensajes en las colas, mediante una operación de transferencia y, normalmente, los recuperan mediante una operación de obtención. Las colas, que pueden ser locales o remotas, las gestionan los gestores de colas. Los datos de configuración se almacenan en un registro.

La estructura de objetos MQSeries Everyplace se puede observar en la pantalla siguiente desde la herramienta de gestión MQe\_Explorer:

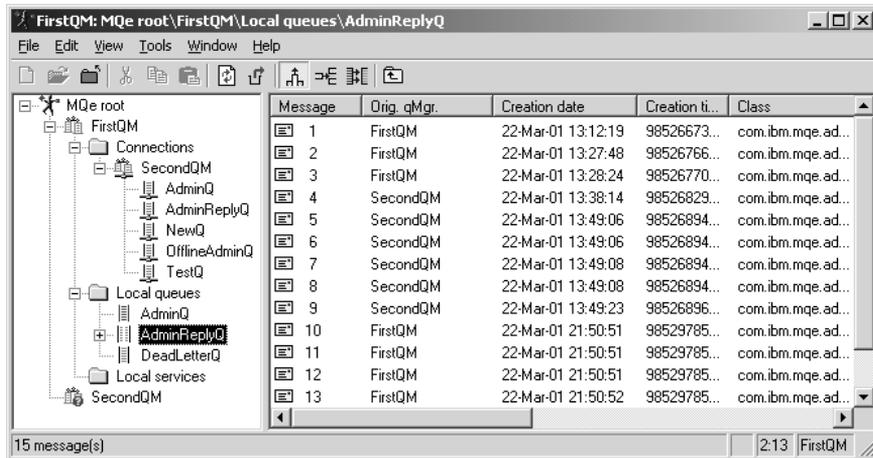


Figura 3. Objetos de MQSeries Everyplace que muestra MQe\_Explorer

En el panel de la izquierda los objetos se muestran en una estructura de árbol. Debajo de la raíz MQe root hay dos gestores de colas, ExampleQM y OfficeQM. La estructura de OfficeQM se ha ampliado para mostrar las carpetas de *Connections*, *Local queues* y *Local services*. Al ampliar *Connections* se muestra una definición de conexión del gestor de colas OfficeQM al gestor de colas ExampleQM. El gestor de colas remotas, OfficeQM, tiene definiciones para dos de las colas de ExampleQM, AdminQ y AdminReplyQ. La ampliación de *Local queues* para OfficeQM muestra cinco colas locales, una de las cuales está seleccionada y cuyo contenido se muestra en el panel situado a la derecha.

## conceptos del producto

La comprensión de MQSeries Everyplace requiere conocer la naturaleza de los mensajes, las colas, los gestores de colas y las conexiones. Estos conceptos se describen en los apartados siguientes.

---

### Objetos de mensaje

Los objetos de mensaje de MQSeries Everyplace presentan diferencias fundamentales en relación con los mensajes para los que ofrece soporte el servicio de mensajería de MQSeries. En MQSeries, un mensaje es una matriz de bytes, dividida en una cabecera de mensaje y un texto de mensaje. La cabecera de mensaje se comprende y contiene información importante como, por ejemplo, la identidad de la cola de respuestas, el gestor de colas de respuestas, el ID de mensaje y el ID de correlación. El texto de mensaje no se comprende.

Por el contrario, los mensajes de MQSeries Everyplace son *objetos de mensaje* que se heredan a partir de un objeto MQSeries Everyplace al que se conoce como *objeto de campos*. Estos mensajes no tienen ningún concepto de cabecera o de texto de mensaje, pero tienen propiedades y métodos. Para comprender el objeto de mensaje en primer lugar se debe conocer el objeto de campos predecesor.

Los objetos de campos, que se utilizan comúnmente en MQSeries Everyplace, son una acumulación de *campos*, en la que un campo consta de un nombre, un tipo de datos y los datos propiamente dichos. Los nombres de campos son series de caracteres ASCII (con la excepción de un número de caracteres reservados) de longitud ilimitada.

Los objetos de campos tienen un *tipo* y éste es una serie de caracteres abreviada que corresponde al nombre de clase de objeto de programación. Los tipos de campos pueden ser:

Tabla 3. Tipos de objeto de campos

Tipo	Descripción
<b>ASCII</b>	Serie de caracteres o una matriz dinámica de series de caracteres ASCII
<b>Booleano</b>	Valor
<b>Byte</b>	Matriz fija o una matriz dinámica de valores de byte
<b>Coma flotante doble</b>	Valor, matriz fija o una matriz dinámica de valores de coma flotante doble
<b>Campos</b>	Objeto o una matriz dinámica de objetos de campos (con lo cual se da soporte a objetos de campos anidados)
<b>Coma flotante</b>	Valor, matriz fija o una matriz dinámica de valores de coma flotante
<b>Entero</b>	Valor (4 bytes), matriz fija o una matriz dinámica de enteros
<b>Entero largo</b>	Valor (8 bytes), matriz fija o una matriz dinámica de enteros largos

Tabla 3. Tipos de objeto de campos (continuación)

Tipo	Descripción
Objeto	Valor
Entero corto	Valor (2 bytes), matriz fija o una matriz dinámica de enteros cortos
UNICODE	Serie de caracteres o una matriz dinámica de series de caracteres UNICODE

Los objetos de campos ofrecen soporte para distintas invocaciones de método como, por ejemplo:

- Enumeración de campos
- Copia de un campo o campos integrantes (con la opción de sustitución)
- Comparación de objetos de campos
- Transferencia y obtención de los campos integrantes
- Inspección de objetos de campos (por ejemplo, "¿está contenido un objeto integrante en el objeto?")
- Vuelco y restauración desde una matriz de bytes

Los métodos de vuelco (dump) y restauración (restore) son sumamente importantes. Por ejemplo, se invoca al método de vuelco para proporcionar los datos para la transmisión de un objeto de mensaje a través de un enlace, y el método de restauración permite volver a crear dicho objeto en el receptor. De este modo, cada objeto es responsable de su propio formato de transmisión y se pueden alterar temporalmente los métodos por omisión facilitados para implementar el comportamiento personalizado. Ambos métodos se utilizan de forma parecida al guardar o restaurar objetos de mensaje de las colas. Alterar temporalmente estos métodos puede implementar un comportamiento muy diferente. Por ejemplo, un objeto de mensaje puede consultar una base de datos en el momento del vuelco para extraer su valor antes de la transmisión.

En la tabla siguiente se muestran algunas de las propiedades de los objetos de campos y sus campos integrantes:

Tabla 4. Objetos de campos y propiedades de los campos integrantes

Propiedad	Disponibilidad	
	Objetos de campos	Campos
Objeto de atributo asociado	Opcional	
Objetos integrantes	sí	
Oculto		sí
Nombre		sí
Tipo	sí	sí
Valor		sí

## objetos de mensaje

Dos propiedades a tener en cuenta son la propiedad *oculta*, que permite ignorar un campo cuando se desea realizar comparaciones, y la posibilidad de *asociar un objeto de atributos* con un objeto de campos.

Los *objetos de atributo* son la base del modelo de seguridad de MQSeries Everyplace y permiten el acceso selectivo y la protección del contenido. Sus propiedades más importantes son las siguientes:

Tabla 5. Propiedades de los objetos de atributo

Propiedad	Descripción
<b>Autenticación:</b>	Controla el acceso
<b>Cifrado:</b>	Protege el contenido cuando se realiza un vuelco del objeto (y permite su restauración)
<b>Compresión:</b>	Disminuye los requisitos de memoria (tanto en la transmisión como en el almacenamiento)
<b>Norma:</b>	Controla las operaciones permitidas

Los objetos de mensaje heredan de los objetos de campos, por lo que incluyen todas las posibilidades anteriores, además de poder disponer de objetos de atributo asociados en el nivel de objeto de campos de componente y de mensaje. Por otra parte, los objetos de mensaje incluyen un UID (identificador exclusivo) que genera MQSeries Everyplace. Este UID identifica de modo exclusivo cada objeto de mensaje individual en toda la red MQSeries Everyplace y se crea a partir de:

- **El nombre** del gestor de colas de origen (que ha añadido el gestor de colas cuando ha recibido el objeto). Este nombre debe ser mundialmente exclusivo.
- **La hora** en que se ha creado el objeto de mensaje (se añade al crearlo).

Por este motivo, un objeto de mensaje tiene las propiedades siguientes, además de las heredadas del objeto de campos predecesor:

Tabla 6. Propiedades de los objetos de mensaje

Propiedad	Descripción
<b>Gestor de colas de origen</b>	Nombre del gestor de colas que ha recibido primero el nuevo mensaje de la aplicación
<b>Hora de creación</b>	La hora en que la aplicación ha creado el objeto de mensaje
<b>UID</b>	Identificador exclusivo de MQSeries Everyplace

No es necesario ningún otro tipo de información en un mensaje que vaya dirigido a otro gestor de colas MQSeries Everyplace, aunque es muy probable que se incluyan otros campos. Estos campos adicionales pueden ser:

- Campos que añade MQSeries Everyplace para reflejar el estado actual
- Campos asociados a una subclase de mensaje determinada
- Campos de personalización asociados a una instancia de objeto de mensaje

MQSeries Everyplace añade campos a un objeto de mensaje (y posteriormente los elimina) con el fin de implementar operaciones de gestión de colas y mensajería. Por ejemplo, cuando un mensaje está en una cola, se puede consultar para obtener información sobre cuándo se ha transferido a dicha cola. (Se puede hacer utilizando una norma de colas que se explica más adelante). Cuando se envía un mensaje entre gestores de colas, se puede añadir un campo de nuevo envío para indicar que se está realizando una retransmisión de los datos. Hay muchos otros ejemplos.

Los mensajes habituales basados en aplicaciones son instancias de algún sucesor de la clase de objeto de mensaje base y, por lo tanto, contienen campos adicionales según su finalidad (por ejemplo, en número de factura). De estos campos adicionales, algunos son genéricos o comunes de varias aplicaciones como, por ejemplo, el nombre del gestor de la cola de respuestas. En reconocimiento de esto, MQSeries Everyplace ofrece soporte para los campos que se pueden incluir en los mensajes, entre los que se incluyen:

*Tabla 7. Campos de objetos de mensaje a los que se da soporte*

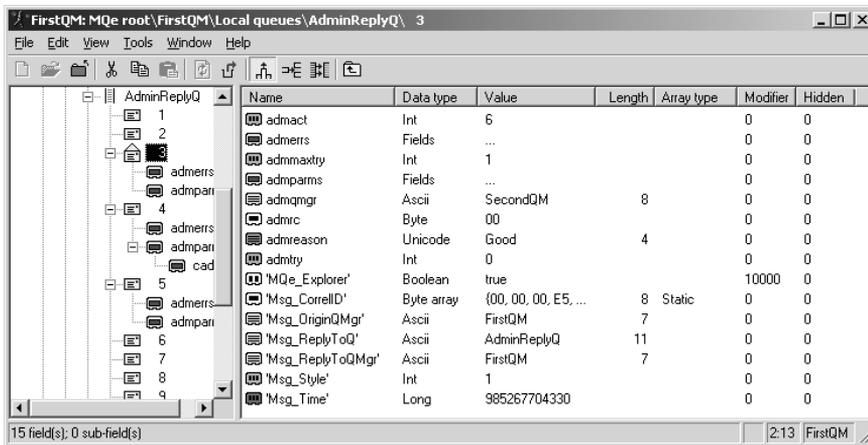
Nombre de campo	Uso
<b>Action</b>	Lo utiliza la administración para indicar acciones como, por ejemplo, consulta, creación o supresión
<b>Correlation ID</b>	Serie de bytes que se utiliza normalmente para correlacionar una respuesta con el mensaje original
<b>Errors</b>	Lo utiliza la administración para devolver información de error
<b>ExpireTime</b>	Hora después de la cual se puede suprimir el mensaje (incluso si no se ha entregado)
<b>Lock ID</b>	La clave necesaria para desbloquear un mensaje
<b>Message ID</b>	Identificador exclusivo de un mensaje
<b>Originating queue manager</b>	El nombre del gestor de colas que ha enviado el mensaje
<b>Parameters</b>	Lo utiliza la administración para pasar detalles de administración
<b>Priority</b>	Prioridad relativa del orden de transmisión de mensajes
<b>Reason</b>	Lo utiliza la administración para devolver información de error
<b>Reply-to queue</b>	El nombre de la cola a la que debe dirigirse una respuesta al mensaje
<b>Reply-to queue manager</b>	El nombre del gestor de colas al que debe dirigirse una respuesta al mensaje
<b>Resend</b>	Indica que el mensaje es un nuevo envío de un mensaje anterior
<b>Return code</b>	Lo utiliza la administración para devolver el estado de una operación de administración
<b>Style</b>	Distingue los mandatos de petición/respuesta, entre otros
<b>Wrap message</b>	Mensaje empaquetado para asegurar la protección de los datos

## objetos de mensaje

En todos los casos se dispone de una constante definida que permite transportar el campo en un solo byte. Para algunos campos existen otras disposiciones, por ejemplo, la prioridad (si está disponible) afecta al orden en el que se transmiten los mensajes, el ID de correlación activa la indexación de una cola con esos valores de campo que permiten una rápida recuperación y la hora de caducidad activa la caducidad del mensaje, etc.

El panel de administración que se muestra en la Figura 4 ilustra la estructura de un mensaje. Tenga en cuenta que la utilización de campos anidados significa que puede haber jerarquías de datos complicadas en los mensajes y que las herramientas de administración (y las aplicaciones) pueden profundizar en la estructura auto-descriptiva (siempre que no se haya inhibido a través de los valores de seguridad).

Los objetos de mensaje utilizan los métodos de restauración (restore) y vuelco (dump) subyacentes cuando se serializan para o desde una matriz de bytes para la transmisión o el almacenamiento en cola. Por omisión, para ahorrar espacio, esto no hace fluir la definición de clase asociada. En caso de que fuera necesario, grabar encima del vuelco o la restauración con métodos de serialización estándar puede hacer fluir la definición de clase.



Name	Data type	Value	Length	Array type	Modifier	Hidden
admact	Int	6			0	0
admerrs	Fields	...			0	0
admmaxtry	Int	1			0	0
admparms	Fields	...			0	0
admqlmgr	Ascii	SecondQM	8		0	0
admrc	Byte	00			0	0
admreason	Unicode	Good	4		0	0
adntry	Int	0			0	0
'MQE_Explorer'	Boolean	true			10000	0
'Msg_CorrelID'	Byte array	{00, 00, 00, E5, ...	8	Static	0	0
'Msg_OriginQMgr'	Ascii	FirstQM	7		0	0
'Msg_ReplyToQ'	Ascii	AdminReplyQ	11		0	0
'Msg_ReplyToQMgr'	Ascii	FirstQM	7		0	0
'Msg_Style'	Int	1			0	0
'Msg_Time'	Long	985267704330			0	0

Figura 4. Estructura de mensaje visualizada con MQE\_Explorer

No obstante, por lo general, la clase está disponible independientemente en cada uno de los gestores de colas en los que se va a convertir en instancia el objeto de mensaje. Si fuera necesario convertir en instancia un objeto (por ejemplo, para almacenamiento intermedio durante la transmisión) y se sabe que el archivo de clase no está disponible, se puede empaquetar el mensaje en otra clase que, normalmente, es la clase de objeto de mensaje por omisión. Esta técnica también resulta útil cuando se utiliza un objeto de atributos para proteger el mensaje. La presencia de este tipo de objeto de atributos significa que no se puede acceder al contenido del mensaje sin las claves de seguridad necesarias. Sin embargo, empaquetar este tipo de mensaje

permite que el mensaje del kernel permanezca totalmente protegido y el mensaje empaquetado se puede volcar y restaurar según se desee.

El método de vuelco de objetos de mensaje por omisión se ha mejorado y ahora el tamaño de la serie de bytes generada es menor, con lo que el almacenamiento y la transmisión de los mensajes resultan más eficaces.

## Formato de los datos de las operaciones de vuelco

Por omisión, el formato de los datos de las operaciones de vuelco codifica los campos de esta forma:

```
{Longitud Identificador Barrera {Datos}} {Longitud Identificador Barrera {Datos}} {
```

donde:

- *Datos*: es el valor de los datos. Se comprimen los enteros y se suprimen los 0 (ceros) y las F iniciales. Los valores booleanos no tienen bytes de datos asociados
- *Barrera*: un byte especial que marca el límite entre el identificador y el elemento de datos opcional. Este byte también indica el tipo del elemento de datos
- *Identificador*: contiene el nombre del campo en una serie de bytes ASCII de longitud variable que termina con un byte final
- *Longitud*: indica la longitud del campo de datos. Se utiliza un número variable que oscila entre 1 y 4 bytes. El primer byte tiene reservados los dos primeros bits para indicar la longitud del campo de longitud. Se da soporte a longitudes comprendidas en un rango entre 0 y 1.073.741.823

El resultado es una corriente de datos extremadamente compacta. Si se comprimen los datos se puede obtener un ahorro de espacio adicional. La compresión de tipo XOR con una corriente de bytes anterior puede generar buenos resultados, pero debido a la naturaleza variable de estos campos y a que el orden de los mismos puede variar, una simple compresión XOR no siempre generará los resultados deseados. MQSeries Everyplace incluye una compresión XOR inteligente, que funciona de campo en campo y que, con toda probabilidad, mejorará la compresión.

---

## Colas

Generalmente, las colas se utilizan para contener los objetos que están a la espera de que los programas de aplicación los supriman. Al igual que los mensajes, las colas también se derivan de los objetos de campos. Normalmente no está permitido el acceso directo de las aplicaciones al objeto de cola. En su lugar, el gestor de colas actúa como intermediario entre los programas de aplicación y las colas. Las colas se identifican por el nombre y éste puede ser una serie de caracteres ASCII de longitud ilimitada<sup>3</sup> pero debe ser exclusivo en un gestor de colas determinado. MQSeries Everyplace da soporte a diferentes tipos de colas:

### Colas locales

---

3. Para permitir las operaciones entre diferentes sistemas se recomienda seguir los convenios de denominación de MQSeries, incluida una longitud máxima de 48 caracteres. La longitud también puede estar limitada por el sistema de archivos que se esté utilizando.

Las colas locales las utilizan las aplicaciones para almacenar mensajes de forma segura y fiable. El mensaje almacenado se correlaciona en el almacenamiento permanente a través de un adaptador, cola a cola. Con MQSeries Everyplace se suministra un rango de adaptadores y se pueden grabar u obtener otros de otras fuentes. El adaptador estándar se conoce como MQeDiskFieldsAdapter, que correlaciona colas en el sistema de archivos local e implementa la entrega asegurada. Otro adaptador, MQeReducedDiskFieldsAdapter, también correlaciona colas en el sistema de archivos, pero cambia la rapidez del rendimiento por una dependencia del sistema operativo que perdura lo suficiente para vaciar sus almacenamientos intermedios en el subsistema de discos físicos. Otro adaptador, MQeMemoryFieldsAdapter, correlaciona colas en la memoria. Aunque ofrece el rendimiento de colas más rápido, este adaptador presenta la característica de que los mensajes no perduran cuando se efectúa un reinicio del sistema operativo o del gestor de colas. Creando el adaptador adecuado, los mensajes se pueden almacenar en cualquier lugar, cola a cola. Una base de datos relacional o un CD grabable son algunos ejemplos de soportes de almacenamiento. Hay adaptadores que se benefician del sistema de archivos duplicado del controlador de tienda de venta al por menor IBM 4690 o que aprovechan DB2 para el almacenamiento en cola.

Las colas locales se pueden utilizar tanto en línea como fuera de línea (conectado o no a una red). Las colas también pueden tener atributos de seguridad establecidos, de forma muy parecida a la protección de objetos de campos y mensajes con el objeto de atributos. La seguridad de colas se trata en el apartado "Seguridad" en la página 43. El acceso a los mensajes de las colas locales siempre es síncrono, lo que significa que la aplicación espera hasta que MQSeries Everyplace vuelve después de finalizar la operación de transferencia u obtención.

### Colas remotas

Las colas remotas son referencias locales a las colas que residen en un gestor de colas remotas. La referencia local tiene el mismo nombre que la cola de destino, pero la definición de la cola remota identifica al gestor de colas propietario de la cola real. Las colas remotas también tienen propiedades relacionadas con el acceso como, por ejemplo, la modalidad de acceso (síncrona o asíncrona), todas las características de seguridad y las opciones de transmisión.

MQSeries Everyplace puede establecer las colas remotas automáticamente. Si se intenta acceder (por ejemplo, para enviar un mensaje) a una cola de otro gestor de colas, MQSeries Everyplace busca la definición de la cola remota. Si existe alguna, se utiliza, pero si no es así, se lleva a cabo la *detección de la cola*. Se detectan las características (autenticación, cifrado y compresión) y se crea la definición de una cola remota. Este tipo de detección de cola depende de si se puede acceder al destino. Si el destino no está accesible, se debe proporcionar una definición remota utilizando otro procedimiento. Cuando se produce la detección de la cola, MQSeries Everyplace establece la modalidad de acceso en síncrona, puesto que se sabe que la cola está disponible de modo síncrono.

Las colas remotas *síncronas* son aquellas a las que sólo se puede acceder cuando se está conectado a una red que tenga una vía de comunicaciones al gestor de colas propietario. Si la red no está establecida, las operaciones de transferencia, obtención y examen (consulte el apartado “Operaciones de los gestores de colas” en la página 34) generan una excepción. La cola controla los permisos de acceso y los requisitos de seguridad necesarios para acceder a la misma. La aplicación es la responsable de manejar los errores o los reintentos durante el envío o la recepción de mensajes puesto que, en tal caso, MQSeries Everyplace deja de ser responsable de la entrega asegurada de una sola vez.

Las colas remotas *asíncronas* son las que envían mensajes a colas remotas, pero no pueden recuperar mensajes de modo remoto. Si se establece una conexión de red, los mensajes se envían al gestor de colas propietario y a la cola. Sin embargo, si la red no está conectada, los mensajes se almacenan localmente hasta que se establece una conexión de red y se transmiten, lo que permite que las aplicaciones puedan operar en la cola cuando el dispositivo está fuera de línea. Por lo tanto, estas colas tienen un adaptador que realiza la correlación con un almacén de mensajes con el fin de que los mensajes puedan almacenarse temporalmente en el gestor de colas emisor en espera de la transmisión.

### **Colas para almacenar y reenviar**

Las colas de este tipo almacenan mensajes asociados a uno o más destinos de gestor de colas de destino. Tienen dos utilizaciones principales. La primera consiste en habilitar el almacenamiento intermedio de mensajes en una red, es decir, que puedan continuar de modo escalonado hacia su destino (una función de envío). La segunda utilización es contener los mensajes a la espera de la recopilación (consulte también el apartado referente a las colas del servidor inicial).

Las colas para almacenar y reenviar están asociadas a un conjunto de nombres de gestores de colas para los que contienen mensajes. Se hace referencia a los mismos como gestores de colas de destino. Los mensajes dirigidos a uno de estos gestores de colas de destino se colocan en la cola para almacenar y reenviar adecuada. Además, la cola para almacenar y reenviar también puede tener un nombre de gestor de colas de envío establecido, en cuyo caso, la cola intenta enviar todos sus mensajes al gestor de colas indicado. Si no se ha establecido ningún nombre, la cola simplemente contiene los mensajes.

Por lo general (aunque no necesariamente), este tipo de cola se define en un servidor o pasarela. Puede haber varias colas para almacenar y reenviar en un solo gestor de colas, pero los nombres de destino no se pueden duplicar. El contenido de una cola para almacenar y reenviar no está disponible para los programas de aplicación. Del mismo modo, una aplicación de envío de mensajes ignora totalmente la presencia o la función de las colas para almacenar y reenviar en la transmisión de mensajes.

Los mensajes de las colas para almacenar y reenviar no están disponibles para las aplicaciones.

### Colas del servidor inicial

Las colas remotas y las colas para almacenar y reenviar transfieren mensajes a través de la red, es decir, son las colas emisoras que inician la transmisión. Sin embargo, las colas del servidor inicial permiten extraer mensajes de una cola remota. Una definición de cola del servidor inicial identifica una cola para almacenar y reenviar en un gestor de colas remotas. A continuación, la cola del servidor inicial extrae de la cola para almacenar y reenviar todos los mensajes destinados al gestor de colas local de la cola del servidor inicial. Un gestor de colas puede contener varias definiciones de colas del servidor inicial, asociando cada una de ellas a una cola para almacenar y reenviar diferente.

Normalmente, las colas del servidor inicial residen en un dispositivo y se establecen para extraer mensajes de un servidor cada vez que el dispositivo se conecte a la red. Cuando la cola del servidor inicial extrae un mensaje del servidor, el mensaje se coloca en la cola local de destino correcta. De este modo, la cola del servidor inicial en sí misma no tiene ningún mensaje al que puedan acceder las aplicaciones. El método de extracción de mensajes del servidor puede resultar más eficaz en cuanto a flujos de red que el método en que el servidor transfiere los mensajes. Esto es debido a que la cola del servidor inicial utiliza el acuse de recibo del primer mensaje como petición del mensaje siguiente (si lo hay), mientras que cuando el servidor transfiere los mensajes es necesaria una petición/respuesta para enviar el mensaje a una segunda petición/respuesta para el flujo de confirmación. Una cola del servidor inicial normalmente tiene establecido un intervalo de sondeo que hace que se compruebe la existencia de mensajes pendientes en el servidor mientras está conectado a la red. El intervalo de sondeo es una opción de configuración de las tareas administrativas. Las colas del servidor inicial tienen una función importante cuando se habilitan clientes para recibir mensajes a través de canales cliente-servidor. La naturaleza de la conexión cliente-servidor implica que los servidores no pueden iniciar la transferencia de datos.<sup>4</sup>

Los mensajes de las colas del servidor inicial no están disponibles para las aplicaciones.

### Colas de administración

Las colas de administración son el mecanismo a través del cual se configuran los gestores de colas (y sus objetos asociados), local o remotamente. La clase de mensaje de administración pertinente procesa los mensajes que se envían a la cola de administración y, a continuación, si fuera necesario, se envía una respuesta a la aplicación de origen. Este tema se trata con más detalle en el apartado “Administración” en la página 37.

### Colas puente de MQSeries

Esta es una cola remota especializada que tiene la definición en una pasarela y la cola de destino en un gestor de colas MQSeries. Este tipo de cola proporciona un puente entre los entornos MQSeries Everyplace y MQSeries. Se utilizan transformadores para realizar cualquier cambio de formato necesario en los datos o en los mensajes. Con MQSeries Everyplace se

---

4. La alternativa consiste en utilizar canales de igual a igual o configurar gestores de colas con posibilidades de cliente y servidor.

proporciona un transformador básico que los programadores deberán personalizar para adaptarlo a sus propios requisitos.

MQSeries Everyplace almacena los datos de forma segura en las colas, y se asegura, dependiendo del adaptador, de que los mensajes se graben físicamente en el soporte de almacenamiento y no queden simplemente en el almacenamiento intermedio del sistema operativo. Sin embargo, MQSeries Everyplace no anota los cambios en mensajes y colas. En el caso de que se produzca una anomalía del soporte de almacenamiento y sea necesaria la recuperación, deberán utilizarse soluciones de hardware como, por ejemplo, los sistemas de discos RAID. De forma alternativa, deberá correlacionarse la cola con almacenamiento recuperable como, por ejemplo, con determinados subsistemas de base de datos.

En MQSeries Everyplace no es necesario que un gestor de colas tenga colas definidas. Sin embargo, se dispone de cuatro colas del sistema, si son necesarias:

- **AdminQ:** una cola necesaria para la recepción de mensajes de administración
- **AdminReplyQ:** una cola que se utiliza opcionalmente para recibir respuestas a los mensajes de administración
- **DeadLetterQ:** una cola que se utiliza para almacenar los mensajes que no se pueden entregar
- **SYSTEM.DEFAULT.LOCAL.QUEUE:** una cola que comparte un nombre común con la cola del sistema obligatoria en los servidores MQSeries

En la tabla siguiente se muestran las propiedades de las colas. Sin embargo, ha de tener en cuenta que no todas las propiedades que se muestran son aplicables a todos los tipos de colas:

*Tabla 8. Propiedades de las colas*

Propiedad	Descripción
Admin_Class	La clase de la cola
Admin_Name	El nombre ASCII de la cola
Queue_Active	Indica que la cola está activa
Queue_AttRule	La clase de norma que controla las operaciones de seguridad
Queue_Authenticator	La clase de autenticador
Queue_BridgeName	Es propietario de un nombre del puente de MQSeries
Queue_ClientConnection	El nombre de la conexión del cliente
Queue_CloseIdle	El cierre del componente de transporte después de transmitir todos los mensajes
Queue_CreationDate	La fecha en la que se ha creado la cola
Queue_Compressor	La clase del compresor
Queue_Cryptor	La clase del método de cifrado
Queue_CurrentSize	El número de mensajes que hay en la cola
Queue_Description	La descripción UNICODE

Tabla 8. Propiedades de las colas (continuación)

Propiedad	Descripción
Queue_Expiry	La hora de caducidad de los mensajes
Queue_FileDesc	La ubicación y el adaptador de la cola
Queue_MaxMsgSize	La longitud máxima permitida para los mensajes de la cola
Queue_MaxQSize	El número máximo de mensajes permitido
Queue_Mode	Síncrona o asíncrona
Queue_MQQMgr	El proxy del gestor de colas MQSeries
Queue_Priority	La prioridad que se utilizará con los mensajes (a menos que la altere temporalmente mediante un valor del mensaje)
Queue_QAliasNameList	Los nombres alternativos de la cola
Queue_QMgrName	El gestor de colas propietario de la cola real
Queue_QMgrNameList	Los destinos del gestor de colas
Queue_RemoteQName	El nombre del campo MQSeries remoto
Queue_Rule	La clase de norma para las operaciones de cola
Queue_QTimerInterval	El retardo antes de procesar los mensajes pendientes
Queue_TargetRegistry	El tipo de registro de la cola
Queue_Transporter	La clase del transportador
Queue_TransporterXOR	El componente de transporte que se va a utilizar para la compresión XOR
Queue_Transformer	La clase del transformador

Las funciones de administración se utilizan para crear y eliminar colas y para consultar o modificar sus propiedades.

En los paneles de MQe\_Explorer siguientes se muestran dos de los cuatro separadores de propiedades que describen las propiedades de una cola local. Los campos inhabilitados indican que una propiedad determinada no se aplica a una cola de dicha clase.

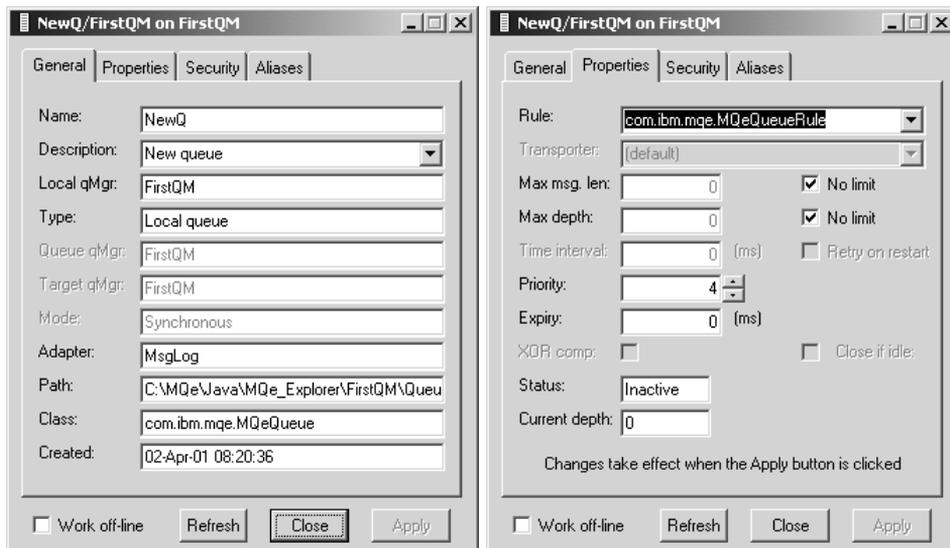


Figura 5. Propiedades de las colas locales visualizadas con MQE\_Explorer

Las colas no necesariamente deben utilizarse como un almacenamiento de mensajes. Se pueden utilizar colas subclasificadas en ejemplos de aplicaciones de control de procesos como, por ejemplo, cuando el objeto de cola controla directamente una válvula. Un mensaje de la clase correcta haría que se abriera la válvula o que cambiara el volumen del flujo, etc. En este caso, la aplicación no extraería los mensajes de la cola ni realizaría la acción, sino que sería el objeto de cola propiamente dicho el que controlaría la acción. Otras colas podrían, por ejemplo, actualizar las hojas de cálculo o realizar la conversión de texto a lenguaje oral. Las ventajas de esta técnica incluyen los aspectos de seguridad que se continúan aplicando y siguen en vigor, al igual que la mensajería asegurada. De este modo, MQSeries Everyplace continúa garantizando la entrega de mensajes de una sola vez y los métodos de autenticación y cifrado asociados garantizan que sólo el emisor autorizado pueda enviar este tipo de mensajes y que durante su transporte el contenido esté fuertemente protegido. No se permitirá que ninguna aplicación acceda a la cola y ninguna será necesaria.

## Gestores de colas

El gestor de colas MQSeries Everyplace permite que las aplicaciones accedan a los mensajes y las colas y controla los canales. En MQSeries Everyplace Versión 1.2 sólo un gestor de colas puede estar activo en una sola máquina virtual Java. Si hay varias JVM en una máquina, puede haber el mismo número de gestores de colas que de JVM. Los gestores de colas se identifican por el nombre y éste debe ser exclusivo mundialmente<sup>5</sup> y por una serie de caracteres ASCII de longitud ilimitada.<sup>6</sup>

5. Esta limitación no la aplican ni MQSeries Everyplace ni MQSeries, sino que es debida a que si existen nombres de gestores de colas repetidos, los mensajes se pueden entregar a un gestor de colas erróneo.

## gestores de colas

Un gestor de colas que se ha configurado con posibilidades de puente y, por consiguiente, puede intercambiar mensajes con productos distribuidos y de sistema principal MQSeries, se conoce como pasarela.

Los gestores de colas se pueden configurar con o sin colocación en colas locales. Todos los gestores de colas soportan las operaciones de mensajería síncrona. Un gestor de colas que permite la colocación en colas locales también da soporte a la entrega de mensajes asíncrona.

La naturaleza de las definiciones de las colas en el gestor de colas emisor determina la elección de si se utiliza la entrega de mensajes síncrona o asíncrona. Si existe una definición de cola remota de modalidad síncrona para la cola de destino, se utiliza la entrega síncrona. Si existe una definición de cola remota de modalidad asíncrona, se utiliza la entrega asíncrona, en la que la definición proporciona el almacenamiento local mientras los mensajes están a la espera de la transmisión. Si no existe ninguna definición de cola remota, pero existe una cola para almacenar y reenviar que maneja los mensajes para dicho gestor de colas de destino, se utiliza la entrega asíncrona. En este caso, la cola para almacenar y reenviar proporciona almacenamiento local para los mensajes en espera de transmisión. Si no existe ninguna definición de cola, se lleva a cabo la detección de la cola y, si es satisfactoria, el resultado es la mensajería síncrona.

Sin tener en cuenta si se utiliza la mensajería síncrona o asíncrona, MQSeries Everyplace puede utilizar la transmisión directa o indirecta, dependiendo de las definiciones de conexión disponibles (consulte el apartado “Conexiones” en la página 35). La transmisión directa sólo implica dos gestores de colas: el gestor de colas emisor y el gestor de colas de destino. La conexión indirecta implica una serie de gestores de colas, con posibles cambios de protocolo en la ruta. Cuando se utiliza la transmisión indirecta con la mensajería síncrona, el comportamiento de la mensajería no cambia respecto al caso de la transmisión directa. Los gestores de colas intermedios simplemente establecen la conectividad entre el origen y el destino. Sin embargo, cuando se utiliza la transmisión indirecta con la mensajería asíncrona, los intermediarios de la transmisión pueden convertirse en puestos de etapa para el mensaje a medida que éste se desplaza del origen al destino. Esta utilización depende de si se ha definido el almacenamiento de colas intermedias apropiado en los gestores de colas intermedios, en forma de colas para almacenar y reenviar o definiciones de colas remotas adecuadas.

Las entregas de mensajes asíncronas y síncronas tienen diferentes características y consecuencias. En la entrega de mensajes asíncrona, la aplicación pasa el mensaje a MQSeries Everyplace para su entrega en una cola remota. Se efectúa un retorno inmediato a la aplicación. Si el mensaje se puede entregar inmediatamente (o mover a un puesto de etapa apropiado), se envía y, en caso contrario, se retiene localmente. Las normas asociadas al gestor de colas y a las colas pertinentes definen la lógica de reintentos de transmisión (consulte el apartado “Normas” en la página 51). La entrega

---

6. Para permitir las operaciones entre diferentes sistemas se recomienda seguir los convenios de denominación de MQSeries, incluida una longitud máxima de 48 caracteres. La longitud también puede estar limitada por el sistema de archivos que esté utilizando.

asíncrona proporciona la calidad del servicio de entrega asegurada de una sola vez, puesto que el mensaje se pasa a MQSeries Everyplace, que se convierte en el responsable de la entrega.

En la entrega de mensajes síncrona, la aplicación transfiere el mensaje a MQSeries Everyplace para su entrega a la cola remota. MQSeries Everyplace contacta de forma asíncrona con la cola de destino y coloca en ésta el mensaje. Después de la entrega, MQSeries Everyplace regresa a la aplicación. Se efectúa un retorno inmediato a la aplicación. Si el mensaje no se puede entregar, la aplicación emisora recibe inmediatamente una notificación. En el caso de la entrega síncrona, MQSeries no se responsabiliza de la entrega del mensaje.

Las definiciones de cola remota síncrona y asíncrona se pueden establecer libremente a través de una red MQSeries Everyplace. Cuando la red también incluye gestores de colas de mensajería MQSeries (y sus colas asociadas) hay varias restricciones importantes:

1. Un gestor de colas MQSeries que no esté directamente conectado a una pasarela MQSeries Everyplace no puede llevar a cabo tareas de mensajería síncrona (puesto que no se ofrece soporte para mensajería síncrona a través de canales de mensajes MQSeries). Para minimizar las consecuencias de este problema, se ha cambiado la definición de la entrega síncrona en este caso y se ha redefinido como si se tratara de una entrega al gestor de colas MQSeries conectado directamente a la pasarela. Más allá de este gestor de colas, se utiliza la mensajería asíncrona de MQSeries para permitir que el mensaje llegue al final de su trayecto.
2. En MQSeries Everyplace Versión 1.2 sólo se puede establecer una definición de cola remota síncrona para una cola remota de mensajería de MQSeries. Teniendo en cuenta las consecuencias del punto (1) anterior, esto significa que no se puede llevar a cabo la entrega asíncrona (utilizando esta definición) para las colas de un gestor de colas MQSeries conectado directamente a la pasarela. Sin embargo, si se necesita la colocación en colas en este caso, se puede disponer en la red MQSeries Everyplace utilizando un segundo gestor de colas MQSeries Everyplace mediante la utilización de definiciones de colas remotas adecuadas (o colas para almacenar y reenviar).

De este modo, la entrega de mensajes asíncrona significa que la aplicación local pasa el mensaje a MQSeries Everyplace y su entrega posterior desde el gestor de colas locales pasa a ser responsabilidad de MQSeries Everyplace. Esto significa que no es necesario que tanto la red como la aplicación receptora estén disponibles. La aplicación emisora desconoce la hora de la entrega real. La entrega de mensajes síncrona requiere que la red esté ejecutándose pero la aplicación emisora sabe que se ha entregado a la cola de la aplicación receptora. No es necesario que la aplicación receptora esté disponible ni en la entrega asíncrona ni en la síncrona.

MQSeries Everyplace no ofrece variaciones en la calidad del servicio de entrega asegurada de una sola vez para la mensajería asíncrona, puesto que lleva a cabo la mensajería de MQSeries con sus operaciones de permanencia y no permanencia. En MQSeries Everyplace, las relaciones de rendimiento y fiabilidad se pueden establecer a nivel de la cola eligiendo la opción del adaptador de almacenamiento de colas, por ejemplo:

## gestores de colas

- MQeDiskFieldsAdapter garantiza que los datos se hayan grabado satisfactoriamente en el disco antes de que continúe el proceso
- MQeReducedDiskFieldsAdapter garantiza que los datos se hayan pasado al sistema operativo (y que se puedan recuperar) antes de que continúe el proceso
- MQeMemoryFieldsAdapter guarda datos en la memoria

En todos los casos se implementa la entrega asegurada de una sola vez, aunque la recuperación depende del almacén de mensajes elegido. Las relaciones de rendimiento se describen en el SupportPac EP01.

En la Tabla 9 se muestran las propiedades de los gestores de colas.

*Tabla 9. Propiedades de los gestores de colas locales*

Propiedad	Descripción
Admin_Class	La clase del gestor de colas
Admin_Name	El nombre del gestor de colas
QMgr_ChnlAttrRules	Las normas de los atributos de canal
QMgr_ChnlTimeout	El tiempo excedido del canal
QMgr_Description	La descripción UNICODE
QMgr_QueueStore	El adaptador y la ubicación por omisión para las colas
QMgr_Rules	La clase de norma para las operaciones del gestor de colas

En el panel siguiente se muestra uno de los dos separadores de propiedades que describen las propiedades de un gestor de colas:

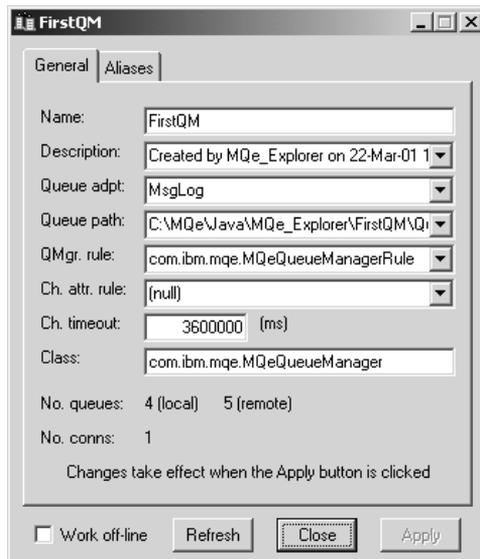


Figura 6. Propiedades del gestor de colas que muestra MQE\_Explorer

## Configuración del gestor de colas

El gestor de colas se ejecuta en un entorno que establece MQSeries Everyplace antes de cargar el gestor de colas. En gestor de colas en sí mismo almacena su información de configuración en su registro (puede obtener información más detallada sobre el mismo en el apartado “El registro” en la página 48). Las colas en sí mismas (que contienen mensajes) se almacenan en almacenamientos de colas.

El entorno MQSeries Everyplace se puede establecer de varios modos, mediante invocaciones a través de API, programas de utilidad que se entregan con MQSeries Everyplace o utilizando herramientas de gestión como, por ejemplo, MQE\_Explorer. Muchas de estas herramientas capturan los parámetros de entorno en un archivo de inicialización, pero es totalmente opcional. Los parámetros de entorno clave se muestran en la tabla siguiente. Los nombres de sección que se utilizan siguen los convenios que se usan para representar esta información en los archivos de inicialización:

Tabla 10.

Nombre de sección	Propiedad	Descripción
[Alias]	(definiciones de alias de clase)	Pueden utilizarse nombres de alias siempre que se necesiten nombres de clase
[ChannelManager]	MaxChannels	Número máximo de canales cliente-servidor simultáneos permitidos

## gestores de colas

Tabla 10. (continuación)

Nombre de sección	Propiedad	Descripción
[Listener]	Network	El adaptador y el número de puerto que están a la escucha de las peticiones de conexión de canal cliente/servidor de entrada
	Listen	El adaptador que se va a utilizar para manejar las conexiones resultantes
	TimeInterval	El tiempo excedido del canal cliente/servidor
[MQBridge]	(parámetros de inicialización para el puente)	
[MQe_Explorer]	(información de direccionamiento guardada)	MQe_Explorer guarda información que describe el modo en que se pueden realizar las conexiones de entrada para este gestor de colas
[Permission]	(mandatos permitidos)	Los mandatos de canal permitidos, las correlaciones de descriptor de archivo y de clase de adaptador
[PreLoad]	clases que se van a cargar al inicializar el gestor de colas	Proporciona un mecanismo para cargar clases de aplicación
[QueueManager]	Name	Nombre del gestor de colas
[Registry]	DirName	La ubicación del registro
	LocalRegType	El tipo de registro (archivo o privado)
	PIN	PIN del registro (o petición de solicitud)
	CertReqPIN	PIN del certificado (o petición de solicitud)
	KeyRingPassword	Contraseña de conjunto de claves (o petición de solicitud)
	CAIPAddrPort	Dirección IP de autorización de certificado

Las cuatro ilustraciones siguientes muestran los separadores más importantes para la creación de un nuevo gestor de colas utilizando MQe\_Explorer. No se presupone ningún entorno previo y no se necesita ningún archivo de inicialización existente. El resultado es un gestor de colas en ejecución, con los datos de configuración guardados en un archivo de inicialización de modo que el gestor de colas se puede reiniciar, simplemente, abriendo dicho archivo.

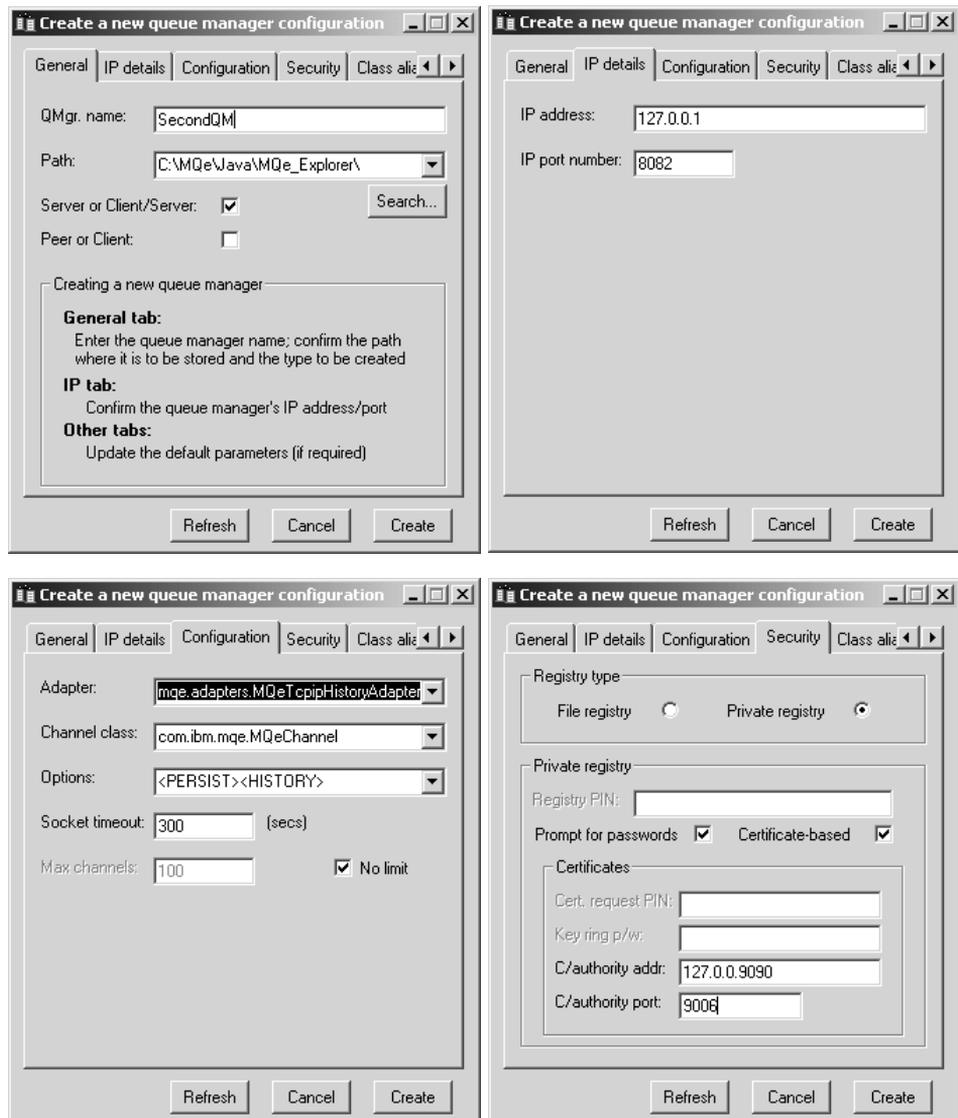


Figura 7. Creación de un gestor de colas con MQe\_Explorer

En realidad, para crear un gestor de colas no es necesario conocer la dirección IP del gestor de colas. Sin embargo, MQe\_Explorer captura la información para utilizarla al configurar otros gestores de colas que se vayan a comunicar con éste. Se necesita el número de puerta para poder escuchar las peticiones de conexión cliente/servidor de entrada.

## gestores de colas

El escucha de canal necesita los datos de configuración del adaptador: el intervalo de tiempo y el número máximo de canales para el gestor de canales. MQe\_Explorer captura el resto de la información tal como se ha indicado anteriormente para habilitar la configuración de otros gestores de colas.

En este ejemplo se ha configurado un registro seguro con autenticación basada en certificado (consulte el apartado “Seguridad” en la página 43). Los PIN y las contraseñas se han inhabilitado puesto que MQe\_Explorer los solicita más adelante, cuando son necesarios.

## Operaciones de los gestores de colas

Los gestores de colas soportan las operaciones de mensajería y, opcionalmente, la gestión de las colas. Las aplicaciones acceden a los mensajes a través de los servicios del gestor de colas utilizando métodos como, por ejemplo, *get*, *put*, *browse*, *wait*, *listen* y *delete*. Muchas de estas operaciones toman un *filtro* como uno de los parámetros. Un filtro es un objeto de campos con el que se buscan coincidencias y para realizar una recuperación selectiva se pueden utilizar cualquiera de los campos del mensaje. Muchas invocaciones de método también incluyen un objeto de atributo que se utilizará para codificar o descodificar un mensaje.

La operación de obtención elimina mensajes de una cola de modo destructivo. Dependiendo de las condiciones que imponga el filtro, los mensajes se recuperan por orden de prioridad y, dentro de éste, en el orden de la hora de llegada a la cola. De este modo, siendo todo lo demás igual, se recuperará el primer mensaje que llegue. La operación de obtención se puede llevar a cabo en uno o en dos pasos. La operación en dos pasos se proporciona para situaciones en las que sea esencial que no exista ninguna posibilidad de pérdida de los mensajes cuando pasen de MQSeries Everyplace a la aplicación. En primer lugar se emite una operación de obtención con un *ID de confirmación* (su valor lo selecciona la aplicación). Esta operación obtiene el mensaje para la aplicación, pero lo oculta en la cola en lugar de suprimirlo inmediatamente. Una operación de confirmación posterior, en la que se especifica el UID original del mensaje, indica que la operación de obtención se ha realizado correctamente y, a continuación, se suprime el mensaje. Si la operación de obtención no se realiza correctamente se puede recuperar el mensaje. Las operaciones de transferencia se comportan de modo similar.

Al especificar el UID, los mensajes se pueden *suprimir* de una cola, sin recuperarlos.

Si se necesita una lectura no destructiva, se pueden *examinar* las colas para buscar los mensajes (opcionalmente bajo el control de un filtro). Al examinar se recuperan todos los objetos de mensajes que coinciden con el filtro, pero se dejan en la cola. También se puede *examinar bajo bloqueo*. Esta función adicional permite bloquear los mensajes coincidentes que hay en la cola. Los mensajes se pueden bloquear individualmente, o en grupos identificados mediante un filtro, y la operación de bloquear devuelve un *ID de bloqueo*. Los mensajes bloqueados se pueden obtener o suprimir únicamente si se proporciona el ID de bloqueo. Una opción de examinar permite devolver los mensajes completos o únicamente los UID.

Las aplicaciones pueden *esperar* durante un tiempo especificado a que los mensajes lleguen a una cola. Opcionalmente, se puede utilizar un filtro para identificar los que interesan y también se puede especificar un *ID de confirmación*. De forma alternativa, las aplicaciones pueden realizar escuchas de sucesos de mensajes MQSeries Everyplace, operación en la que también se puede utilizar opcionalmente un filtro. Cuando llegan los mensajes a una cola se notifica a los escucha.

Las colas están habilitadas para las operaciones de mensajería como se muestra en la Tabla 11

Tabla 11. Operaciones de mensajería en colas MQSeries Everyplace

	Cola local	Cola remota <sup>1</sup>	
		Síncrona	Asíncrona
Examinar (±bloqueo, ±filtro)	Sí	Sí	
Eliminar	Sí	Sí	
Obtener (±filtro)	Sí	Sí	
Escuchar (±filtro)	Sí		
Transferir	Sí	Sí	Sí
Esperar (±filtro)	Sí	Sí	

**Notas:**

1. La operación de espera síncrona en cola remota se implementa mediante un sondeo de la cola remota, por lo que el tiempo de espera real es un múltiplo de la duración del sondeo.
2. <sup>1</sup>El puente de MQSeries Everyplace de MQSeries que se proporciona con MQSeries Everyplace Versión 1.2 sólo ofrece soporte para las operaciones de transferir de modo seguro/no seguro, obtener de modo no seguro y examinar de modo no seguro (sin bloqueo).

## Conexiones

Los objetos de conexión definen la topología y el acceso a través de la red MQSeries Everyplace. Estas definiciones se almacenan localmente en cada gestor de colas. Se crean, modifican y destruyen según las disposiciones de administración estándar.

Generalmente, un objeto de conexión define el acceso a un gestor de colas remotas (y, por este motivo, en ocasiones se hace referencia a los mismos como definiciones de gestor de colas remotas). En la tabla siguiente se indican las propiedades:

Tabla 12. Propiedades de conexión (gestor de colas remotas)

Propiedad	Descripción
Admin_Name	El nombre del gestor de colas
Con_Adapter	El descriptor del archivo del adaptador
Con_AdapterOptions	Las opciones del adaptador (como la historia del uso)
Con_AdapterParm	Los datos ASCII que ha de utilizar un adaptador (como el nombre del servlet)

Tabla 12. Propiedades de conexión (gestor de colas remotas) (continuación)

Propiedad	Descripción
Con_Aliases	Los nombres alternativos de las conexiones/gestores de colas
Con_Channel	El tipo de canal que debe utilizar esta conexión
Con_Description	La descripción UNICODE
Queue_QMgrName	El propietario de la definición

En los paneles de administración siguientes se muestran dos de los separadores asociados con una definición de conexión:

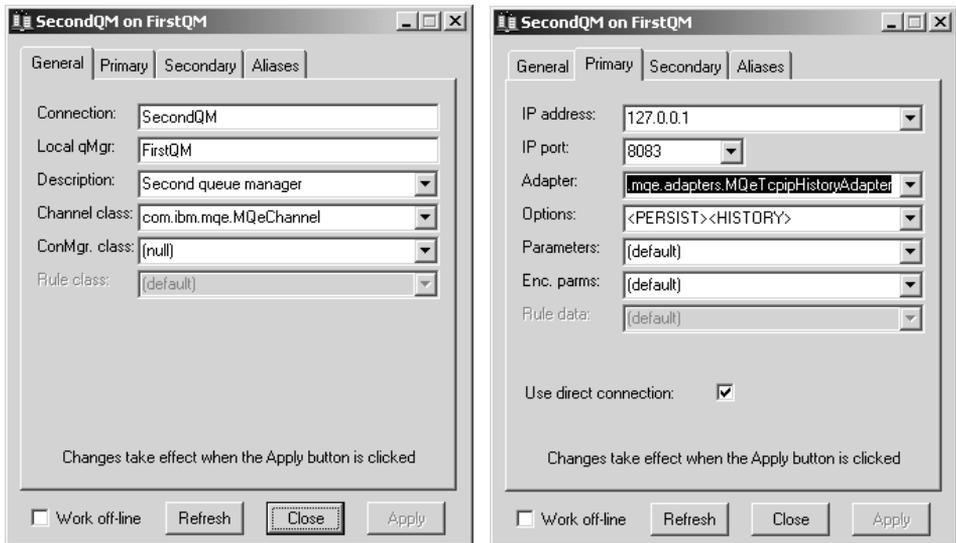


Figura 8. Definición de conexión que se muestra con MQe\_Explorer

Los datos se pueden pasar al adaptador de comunicaciones elegido en formato de opciones, parámetros o parámetros codificados.

Se pueden especificar conexiones indirectas. En este caso, MQSeries Everyplace direcciona la conexión a través de otros gestores de colas (que se pueden encadenar), y puede haber cambios de protocolo en la ruta. Las conexiones indirectas son especialmente útiles para habilitar dispositivos de modo que tengan un único punto de entrada a una red MQSeries Everyplace.

Los objetos de conexión también se utilizan para definir escuchas para canales de igual a igual de entrada.

Como sucede con la mayor parte de los objetos MQSeries, pueden definirse alias para las conexiones. Para definir nombres de alias para el gestor de colas locales, se utiliza una conexión local (definida como una conexión cuyo nombre coincida con el del gestor de colas locales).

## Administración

Las tareas de administración proporcionan funciones para configurar y gestionar los recursos de MQSeries Everyplace como, por ejemplo, las colas y las conexiones. Las funciones relacionadas con los mensajes se consideran responsabilidad de las aplicaciones. Las tareas de administración se habilitan a través de una interfaz que maneja la generación y la recepción de los mensajes de administración y está diseñada de modo que tanto la administración local como la remota se manejen del mismo modo. Las peticiones se envían a la cola de administración del gestor de colas de destino y, si es necesario, se pueden recibir respuestas. Cualquier programa de aplicación MQSeries Everyplace local o remoto puede crear y procesar mensajes de administración, tanto directa como indirectamente, a través de los métodos de ayuda. Los mensajes de administración también se pueden generar de forma indirecta mediante MQe\_Explorer<sup>7</sup>, una herramienta de gestión que proporciona una interfaz gráfica de usuario para la administración del sistema.

La cola de administración no comprende cómo debe administrar los recursos individuales, sino que estos conocimientos se encuentran encapsulados en cada uno de los recursos y en su mensaje de administración correspondiente.

## Mensajes de administración

Los mensajes amplían el objeto de mensajes MQSeries Everyplace base. La Tabla 13 lista las clases de mensajes que se proporcionan para la administración de los recursos de MQSeries Everyplace. Se pueden crear subclases de estos mensajes de administración base para las tareas de administración de otros objetos. Por ejemplo, se puede gestionar un tipo de cola diferente utilizando una subclase de MQeQueueAdminMsg. El puente entre MQSeries Everyplace y MQSeries utiliza subclases de MQeAdminMsg de este modo.

Tabla 13. Clases de mensajes de administración

Clases de mensajes de administración	Uso
MQeAdminMsg	Una clase abstracta que se utiliza como base para todos los mensajes de administración
MQeQueueManagerAdminMsg	Administración de los gestores de colas
MQeQueueAdminMsg	Administración de las colas locales
MQeRemoteQueueAdminMsg	Administración de las colas remotas
MQeAdminQueueAdminMsg	Administración de la cola de administración
MQeHomeServerQueueAdminMsg	Administración de las colas del servidor inicial

7. MQe\_Explorer no está incluido con MQe\_Explorer, pero se puede bajar gratuitamente desde el sitio MQSeries Everyplace en la World Wide Web (<http://www.ibm.com/software/ts/mqseries/everypplace>).

Tabla 13. Clases de mensajes de administración (continuación)

Clases de mensajes de administración	Uso
MQeStoreAndForwardQueueAdminMsg	Administración de las colas para almacenar y reenviar
MQeConnectionAdminMsg	Administración de las conexiones entre gestores de colas
MQeClientConnectionAdminMsg	Administración de un objeto de conexión de cliente, que se utiliza como puente para conectarse a MQSeries
MQeListenerAdminMsg	Administración de un objeto de escucha de colas de transmisión, que se utiliza como puente para recoger mensajes de MQSeries
MQeBridgeAdminMsg	Administración de un puente con MQSeries
MQeMQBridgesAdminMsg	Administración de una lista de puentes MQ
MQeMQMgrProxyAdminMsg	Administración de una representación puente de un gestor de colas MQSeries
MQeMQBridgeQueueAdminMsg	Administración de una cola puente de MQSeries

La estructura de un mensaje de administración depende de su clase individual, esto es, de la naturaleza del recurso que gestiona, y de los detalles de la operación que se ha de llevar a cabo en dicho recurso. Sin embargo, genéricamente la estructura de los mensajes de administración es la que se muestra en la Tabla 14:

Tabla 14. Estructura genérica de un mensaje de administración

Campos de nivel 1	Campos de nivel 2 e inferiores	Uso
Admin_Action		Crear, eliminar, consultar, etc.
Admin_Errors		Objeto de campos principal
	Varios campos	Información detallada basada en los errores individuales
Admin_MaxAttempts		El número máximo de veces que debe intentarse la acción de administración
Admin_Parameters		Objeto de campos principal
	Recurso	El nombre del recurso que se ha de gestionar
	Varios campos	Los datos de parámetros detallados específicos de la clase del mensaje y de la acción
Admin_Reason		El mensaje de texto que indica la razón de la anomalía
Msg_ReplyToQ		El nombre de la cola a la que debe enviarse la respuesta

Tabla 14. Estructura genérica de un mensaje de administración (continuación)

Campos de nivel 1	Campos de nivel 2 e inferiores	Uso
Msg_ReplyToQMgr		El nombre del gestor de colas al que debe enviarse la respuesta
Admin_RC		El código de retorno numérico que indica el resultado
Msg_Style		El mandato o la petición/respuesta
Admin_TargetQMgr		El nombre del gestor de colas propietario del recurso de destino

Se da soporte a tres tipos de mensajes de administración, principalmente mandatos (datagramas) que indican una acción de administración que no requiere una respuesta, peticiones que requieren una respuesta y las respuestas propiamente dichas. La respuesta se crea a partir de una copia del mensaje original, de este modo, el emisor puede añadir campos adicionales que el receptor puede utilizar.

### Administración selectiva

El acceso a las tareas de administración se puede controlar mediante el autenticador de la cola de administración. El autenticador que se suministra considera que todas las aplicaciones locales representan al mismo usuario local y, por lo tanto, permite o prohíbe las tareas de administración a todas ellas. Las aplicaciones de administración remotas se controlan invocando al autenticador que se encuentra en el canal antes de que fluyan los mensajes de administración. De este modo, se puede diferenciar a los diferentes usuarios remotos y habilitarlos o inhabilitarlos individualmente. En todos los casos y para cualquier usuario, las tareas de administración se habilitan o inhabilitan por completo. Si se requiere un control de las tareas de administración más exhaustivo, por ejemplo, si se da acceso a algunas colas a determinados usuarios de administración pero a otros no, entonces es necesario realizar tareas de programación adicionales. Un autenticador más sofisticado puede realizar un seguimiento de los permisos asociados a las identidades de los usuarios, y los mensajes de administración se pueden procesar a continuación basándose en estos permisos (consulte el apartado “Seguridad” en la página 43). También se pueden utilizar las normas asociadas a las colas para permitir o prohibir acciones del mismo modo (consulte el apartado “Normas” en la página 51).

### Supervisión y otras acciones relacionadas

Con frecuencia, la administración va más allá del ámbito de creación y modificación de objetos. Puede incluir la supervisión de un sistema y la información a un operador cuando una cola está llena, o el modo en que se manejan situaciones de error, lo que implica, por ejemplo, la realización de las acciones adecuadas cuando llega un mensaje que es demasiado grande para su cola de destino. MQSeries Everyplace maneja estos aspectos mediante el uso de las normas, que son clases que se invocan cuando hay objetos cuyo estado cambia de manera significativa o cuando surgen

determinadas situaciones de error. Con MQSeries Everyplace se proporciona un conjunto de clases de normas por omisión, pero normalmente las clases se personalizan (consulte el apartado “Normas” en la página 51).

---

### Canales dinámicos

MQSeries Everyplace se comunica con los gestores de colas a través de enlaces lógicos, conocidos como canales dinámicos. Estos canales soportan flujos bidireccionales y el gestor de colas los establece según sea necesario. Tanto la mensajería síncrona como la asíncrona utilizan los mismos canales y el protocolo es exclusivo de MQSeries Everyplace. Este protocolo se puede personalizar basándose en cada mensaje alterando temporalmente el método de vuelco del mensaje. Por el contrario, MQSeries normalmente utiliza canales de cliente para el tráfico síncrono y un par de canales de mensajes para la mensajería asíncrona bidireccional. Los *canales de mensajes del clúster* de MQSeries tienen algunas características similares a las de los canales dinámicos de MQSeries Everyplace, pero existen muchas diferencias importantes.

Un canal dinámico es una conexión lógica entre dos gestores de colas que se ha establecido para enviar o recibir datos. Pueden existir varios canales a la vez, incluso entre las mismas partes. Presentan características como, por ejemplo, la autenticación, el cifrado, la compresión y el protocolo de transporte utilizado. Estas características son accesorias, (se pueden utilizar versiones diferentes en canales diferentes) y, por lo tanto, cada canal tiene sus propios atributos de calidad de servicio en cuanto a:

- **Autentificador:** puede ser nulo o un objeto *autentificador* que efectúe las tareas de autenticación de usuarios o de canales.
- **Canal:** la clase que proporciona los servicios de transporte.
- **Compresor:** puede ser nulo o un objeto *compresor* que puede realizar la compresión y descompresión de los datos.
- **Objeto cifrador:** puede ser nulo o un *objeto cifrador* que puede realizar las tareas de codificación y decodificación.
- **Destino:** el destino de este canal, por ejemplo, SERVER.XYZ.COM

Normalmente, el autentificador sólo se utiliza cuando se define el canal. Los compresores y los objetos cifradores se utilizan normalmente en todos los flujos.

El tipo más sencillo de objeto cifrador es MQeXorCryptor, que codifica los datos enviados realizando una operación OR exclusiva de los datos. La encriptación no es segura, pero modifica los datos de forma que no pueden verse. Por el contrario, MQe3DESCryptor implementa el tipo de cifrado DES triple. El tipo de compresor más sencillo es MQeRleCompressor, que comprime los datos sustituyendo los caracteres que se repiten por un contador. Se proporcionan otros autentificadores, compresores y objetos cifradores que puede consultar en la Tabla 15 en la página 44.

Al definir un canal se utilizan las especificaciones del adaptador de protocolo para determinar los enlaces y protocolos que se utilizarán en un canal determinado. En cada nodo intermedio se buscan las definiciones del canal para resolver el direccionamiento

que se necesita para el siguiente enlace. Cuando ya no haya una definición de avance, el canal habrá finalizado y, llegado este punto, todos los mensajes que fluyan por el canal se pasarán al gestor de colas. De este modo, se pueden aprovechar las definiciones de colas remotas y las colas para almacenar y reenviar intermedias.

El gestor de colas define los canales, según sea necesario, y los programadores o administradores de aplicación no pueden verlos directamente. Los canales enlazan a los gestores de colas entre sí y sus características las negocia una y otra vez MQSeries Everyplace dependiendo de la información que deba fluir. MQSeries Everyplace posee unos componentes de transporte que utilizan los canales para proporcionar comunicaciones en el nivel de las colas. Una vez más, el programador de aplicaciones o el administrador no puede ver directamente estos componentes.

Cuando es necesaria una mensajería asegurada, MQSeries Everyplace entrega los mensajes a la aplicación una vez y, solamente, una vez. Esto lo lleva a cabo asegurándose de que un mensaje ha pasado de un gestor de colas a otro sin problemas y de que se ha recibido un acuse de recibo, antes de eliminar la copia al final de la transmisión. Si se produce una anomalía de comunicaciones, y no se ha recibido un acuse de recibo, se puede volver a transmitir un mensaje (la entrega de una sola vez no implica que la transmisión sea una sola vez), aunque no se entregarán duplicados.

---

## Adaptadores

Los *adaptadores* se utilizan para correlacionar MQSeries Everyplace con las interfaces de dispositivos. Los canales utilizan los adaptadores de protocolo para su ejecución a través de HTTP, TCP/IP nativo, UDP y otros protocolos. Del mismo modo, las colas utilizan los adaptadores de almacenamiento de campos como interfaz con un subsistema de almacenamiento como, por ejemplo, la memoria o el sistema de archivos. Los adaptadores proporcionan un mecanismo para que MQSeries Everyplace pueda ampliar su soporte de dispositivos y utilizar otras versiones.

Un *descriptor de archivo* es una serie de caracteres que se utiliza para identificar, cargar y activar un adaptador.

---

## Gestión de conexiones de marcación

El soporte de redes de marcación para dispositivos se maneja mediante el sistema operativo de los dispositivos. Cuando MQSeries Everyplace intenta utilizar la red desde un dispositivo desconectado porque, por ejemplo, se debe enviar un mensaje, si la pila de red no está activa, el sistema operativo propiamente dicho inicia los servicios de acceso remoto (RAS). Normalmente, esto adopta la forma de un panel que se muestra al usuario y en el que se le ofrece un perfil de conexión de marcación. Hasta que se establece la conexión, el control lo tiene el sistema operativo. Por lo tanto, el usuario debe asegurarse de que estén disponibles los perfiles de conexión de marcación adecuados para que el sistema operativo pueda utilizarlos. No existe un soporte específico para redes de marcación en MQSeries Everyplace Versión 1.2.

---

### Rastreo

El rastreo se habilita ejecutando un programa independiente que efectúa las acciones de rastreo. En MQSeries Everyplace se han intercalado llamadas para el rastreo en situaciones de información, aviso y errores con variantes del sistema y del usuario. Las aplicaciones también puede realizar acciones de rastreo directamente y pueden añadir nuevos mensajes o modificar los mensajes de rastreo existentes. El programa de rastreo de ejemplo que se suministra permite visualizar, imprimir y dirigir a las anotaciones de sucesos los mensajes seleccionados. Se pueden escribir otros programas de rastreo que presenten posibilidades adicionales o se pueden diseñar para que den formato y entreguen la salida de otras formas.

La mayor parte de las excepciones de MQSeries Everyplace se pasan a la aplicación para que las maneje y el manejador de excepciones de la aplicación puede direccionar estas excepciones para su rastreo.

---

### Anotaciones de sucesos

MQSeries Everyplace proporciona mecanismos de anotación de sucesos e interfaces que se pueden utilizar para anotar estados, por ejemplo, "iniciado por el gestor de colas". Se pueden iniciar las anotaciones y, por omisión, grabarlas en un archivo, sin embargo, también se pueden interceptar y dirigir a cualquier otro lugar. Las anotaciones de sucesos de MQSeries Everyplace no anotan los datos de mensajes y no se pueden utilizar para recuperar los mensajes ni las colas.

---

### Entrega de mensajes

Las redes MQSeries Everyplace son gestores de colas conectados y pueden incluir pasarelas. Pueden abarcar muchas redes físicas y direccionar los mensajes entre ellas. En general, proporcionan un acceso síncrono y asíncrono a las colas con un modelo de programación independiente de la ubicación de la cola.

---

### Entrega de mensajes asíncrona

Cuando se transfiere un mensaje a una cola remota de forma asíncrona, el objeto de mensaje se coloca lógicamente en el almacenamiento de seguridad asociado a la definición local de esta cola, a los nombres del gestor de colas de destino y de la cola y a las características de compresor, autenticador y cifrador que coincidan con el destino del mensaje. Se llama al método de vuelco del objeto cuando éste se guarda en el almacenamiento permanente con un formato seguro, según la definición de su cola de destino. El gestor de colas controla la entrega del mensaje. Para realizar el salto siguiente, identifica (o establece) un canal con las características apropiadas al gestor de colas, a continuación, crea (o vuelve a utilizar) un componente de transporte a la cola de destino. El componente de transporte vuelca el objeto y transmite la serie de bytes resultante. Tenga en cuenta que el nombre del gestor de colas de destino y de la cola no forman parte de este flujo de mensaje.

Si es necesario, el mensaje se cifra y comprime a través del canal. Cuando llega al gestor de colas de destino, se descifra y descomprime. Mediante el método de restauración de dicha clase de objeto, se crea un nuevo objeto de mensaje y es este

objeto resultante el que se coloca en la cola de destino. Si el mensaje no ha llegado al gestor de colas de destino, se descifra y descomprime y, a continuación, se coloca en una cola para almacenar y reenviar que posea las características correctas de avance de transmisión. En ambos casos, se conserva en su cola respectiva en un formato seguro, según la definición de su cola de destino.

Una característica de la entrega de mensajes asíncrona es que los mensajes pasan al gestor de colas mediante saltos intermedios, y se van poniendo en cola para su transmisión progresiva. Los mensajes se sacan de las colas intermedias en primer lugar según el orden de prioridad y después, según la secuencia de indicación de la hora.

### Entrega de mensajes síncrona

La entrega de mensajes síncrona es similar a la entrega asíncrona descrita anteriormente, pero la partición del gestor de colas en los saltos intermedios se lleva a cabo en niveles mucho más bajos y con la ayuda de los componentes de transporte y de los canales. Para identificar el enlace siguiente, se establece un canal de uno a otro extremo, utilizando los adaptadores definidos en las especificaciones de protocolo de cada nodo intermedio. Al final del último enlace, cuando ya no existen más descriptores de archivo relevantes, el mensaje se pasa a las capas más altas del gestor de colas para su proceso. De este modo, el nodo emisor no pone el mensaje en cola sino que lo pasa por el canal, mediante saltos intermedios y, a continuación, lo entrega al gestor de colas de destino para que lo coloque en la cola de destino.

El enlace con MQSeries utiliza una cola puente en la pasarela, que convierte el mensaje a un formato MQSeries. Este mecanismo significa que es posible llevar a cabo el tipo de mensajería síncrono de MQSeries Everyplace desde un dispositivo a MQSeries, con el canal dinámico finalizando en la pasarela. Desde la pasarela, y a través de un canal de cliente, el mensaje se entrega en tiempo real a un servidor MQSeries. Para llegar a su destino desde allí, es posible que sea necesario direccionarlo de forma asíncrona a través de los canales de mensajes de MQSeries.

Del mismo modo, un dispositivo que sólo pueda realizar un envío síncrono de mensajes podrá enviar mensajes a una cola MQSeries Everyplace asíncrona, siempre que haya disponible un intermediario adecuado.

---

## Seguridad

MQSeries Everyplace proporciona un conjunto integrado de características de seguridad que protege los datos de los mensajes tanto cuando se mantienen localmente como durante su transferencia.

Las características de seguridad de MQSeries Everyplace proporcionan protección en tres categorías diferentes:

- Seguridad local - protección local de los datos de mensajes (y otros)
- Seguridad basada en la cola - protección de los mensajes entre el gestor de colas de inicio y la cola de destino

## seguridad

- Seguridad a nivel de mensajes - protección a nivel de mensajes cuando los mensajes están entre el iniciador y el receptor

La seguridad local y el nivel de mensajes de MQSeries Everyplace la utiliza internamente MQSeries Everyplace, pero también está a disposición de las aplicaciones MQSeries Everyplace. La seguridad basada en colas MQSeries Everyplace es un servicio interno.

Las tres características de seguridad de MQSeries Everyplace protegen los datos de los mensajes utilizando un atributo (MQAttribute o un descendiente). Según la categoría, el atributo se aplica explícita o implícitamente.

Todo atributo puede contener los objetos siguientes:

- Autentificador
- Cifrador
- Compresor
- Clave
- Nombre de la entidad de destino

Estos objetos se utilizan de forma diferente, dependiendo de la categoría de la característica de seguridad de MQSeries Everyplace, pero en todos los casos, la protección que ofrece la característica de seguridad de MQSeries Everyplace se aplica cuando se invoca el atributo asociado al objeto de mensaje. Esto sucede al invocar un método de vuelco del mensaje MQSeries Everyplace (cuando se utiliza el método encodeData del atributo para, por ejemplo, cifrar y comprimir los datos del mensaje). La característica de seguridad de MQSeries Everyplace deja de proteger al invocar el método de restauración del mensaje MQSeries Everyplace (cuando se utiliza el método decodeData del atributo para, por ejemplo, cifrar y comprimir los datos del mensaje).

Los algoritmos que soporta MQSeries Everyplace Versión 1.2 para autenticación, cifrado y compresión se describen detalladamente en la Tabla 15.

*Tabla 15. Soporte de autenticación, cifrado y compresión*

Función	Algoritmo
Autenticación	WTLS minicertificado
	Validación de identidad de Windows NT/2000, AIX o Solaris
Compresión	LZW
	RLE
Cifrado	DES triple
	DES
	MARS
	RC4
	RC6
	XOR

## Seguridad local en MQSeries Everyplace

La seguridad local protege los datos de los mensajes MQSeries Everyplace (MQeFields o los descendientes de MQeFields) localmente. Esto se consigue creando un atributo con un compresor y un cifrador simétrico adecuados, creando y definiendo una *clave* apropiada (proporcionando una contraseña o una frase de contraseña). La clave está explícitamente asociada al atributo y el atributo está asociado al mensaje MQSeries Everyplace. MQSeries Everyplace proporciona la clase MQeLocalSecure para ayudar a definir la seguridad local, pero en todos los casos es responsabilidad del usuario de la seguridad local (internamente de MQSeries Everyplace o de una aplicación MQSeries Everyplace) establecer un atributo adecuado y gestionar la clave de contraseña o frase de contraseña.

## Seguridad MQSeries Everyplace basada en colas

La seguridad basada en colas se puede aplicar a los mensajes síncronos y asíncronos.

### Seguridad síncrona basada en colas

La seguridad síncrona basada en colas permite que una aplicación deje todos los temas relacionados con la seguridad de los mensajes en manos de MQSeries Everyplace. Las colas tienen características de autenticación, cifrado y compresión que se utilizan para determinar el nivel de seguridad necesario para proteger los flujos de mensajes (y también para su almacenamiento permanente).

Cuando se ha de enviar un mensaje, las características de seguridad de la cola de destino se recuperan del registro local. Si no están presentes en el registro, el gestor de colas intenta detectar las características de la cola de destino en el gestor de colas de destino y las almacena en la antememoria para su uso posterior. Si existe un canal para ese gestor de colas, se utiliza el canal, si no existe, se crea un nuevo canal y, de este modo, se recuperan los atributos de la cola de destino.

Según la calidad de servicio que sea necesaria, los atributos del canal de acceso al gestor de colas de destino se modifican dinámicamente. Esto estará sujeto a las normas que puedan haberse establecido. Normalmente, una norma permite ir actualizando el nivel de seguridad, (por ejemplo, de no tener ninguna protección a tener una ligera protección o de tener una ligera protección a tener una fuerte protección). Si el canal no se puede actualizar o si el nivel de seguridad resulta excesivo (por ejemplo, si no hace falta ninguna protección y el canal implementa un tipo de protección fuerte) se crea un nuevo canal. Existe una agrupación de canales, que se reutilizan siempre que resulta posible, cuyas características se pueden modificar dinámicamente para ajustarlas a las demandas del tráfico. Los canales se destruyen automáticamente cuando no son necesarios. Los mensajes siempre se colocan en colas con el nivel de seguridad que esté definido en las características de la cola de destino.

La autenticación se lleva a cabo a nivel de canales, lo que mantiene en un mínimo la actividad general que genera cada mensaje. La seguridad basada en colas de mensajes síncronos también se utiliza normalmente con objetos cifradores simétricos, ya que de este modo, el proceso de cifrado/descifrado es más rápido. Sin embargo, en estos casos de simetría, MQSeries Everyplace utiliza inicialmente el cifrado asimétrico RSA para proteger los flujos que se necesitan para definir una clave compartida en el

## seguridad

emisor y en el receptor. Pasado ese punto, se utiliza el cifrado simétrico para proteger la confidencialidad del flujo de datos. MQSeries Everyplace dificulta cualquier ataque al cifrado de estos datos pues modifica dinámicamente la clave en cada flujo de canal. MQSeries Everyplace también asegura la integridad del flujo de datos pues genera y añade a los datos el resumen antes de enviarlos, y vuelve a generarlos y validarlos cuando se reciben.

### Seguridad asíncrona basada en colas

La mensajería asíncrona es diferente de la mensajería síncrona que se ha descrito anteriormente ya que no se garantiza que la cola de destino esté accesible en el momento en el que se ejecuta `putMessage`. En este caso, el gestor de colas no puede enviar inmediatamente el mensaje y lo coloca en la cola de transmisión. Sin embargo, está cifrado según las características de la cola de destino. Cuando se puede transmitir, se descifra y se envía a través de un canal con las características adecuadas. De este modo, los mensajes están siempre protegidos, incluso cuando están a la espera de ser transmitidos. La envío de mensajes asíncrona requiere una definición de cola remota para que se puedan determinar las características de la cola de destino.

En el caso de la mensajería asíncrona, no es posible la autenticación entre el origen y el destino. La seguridad a nivel de mensajes se debe utilizar cuando la autenticación es un tema importante, por ejemplo, para que un receptor pueda determinar la identidad del emisor del mensaje (para determinar si es aceptable o establecer una orden de no rechazo) o para que un emisor se asegure de que el mensaje sólo pueda procesarlo el receptor al que va destinado.

La seguridad basada en colas se utiliza al mismo tiempo que la seguridad a nivel de mensajes pero esto no es necesario puesto que los datos del mensaje ya están protegidos.

### Seguridad a nivel de mensajes

La seguridad a nivel de mensajes protege los datos de los mensajes entre la aplicación MQSeries Everyplace de inicio y de recepción.

La seguridad a nivel de mensajes es un servicio de la capa de aplicación que requiere que la aplicación MQSeries Everyplace de inicio configure un atributo a nivel de mensajes y lo proporcione cuando utiliza `putMessage` para transferir el mensaje a una cola de destino. La aplicación receptora debe configurar y pasar un atributo coincidente a nivel de mensajes al gestor de colas receptor para que el atributo pueda estar disponible cuando la aplicación invoque `getMessage` para obtener el mensaje de la cola de destino.

Tal como ocurren con la seguridad local, la seguridad a nivel de mensajes utiliza la aplicación de un atributo en un objeto de mensaje. El gestor de colas de la aplicación de inicio maneja `putMessage` con el método de vuelco (`dump`), el cual utiliza el método `encodeData` del atributo para proteger los datos del mensaje. Cuando la aplicación receptora invoca `getMessage`, su gestor de colas lo maneja con el método de restauración (`restore`), el cual utiliza el método `decodeData` del atributo para recuperar los datos de mensaje originales.

MQSeries Everyplace proporciona dos atributos alternativos para la seguridad a nivel de mensajes:

### **MQeMAttribute**

Se utiliza para las comunicaciones de empresa a empresa en las que la confianza mutua se gestiona fuertemente en la capa de aplicación y no se requiere fiabilidad de terceros. Se pueden utilizar todas las opciones disponibles de compresor y cifrador simétrico que proporciona MQSeries Everyplace. Al igual que con la seguridad local, la clave del atributo debe estar establecida previamente antes de proporcionarla con `putMessage` o `getMessage`. MQeAttribute proporciona un método sencillo y potente para la protección a nivel de mensajes que permite utilizar un cifrado fuerte para proteger la confidencialidad de los mensajes, sin la actividad general que genera la infraestructura de claves públicas (PKI).

### **MQeMTrustAttribute**

Este atributo proporciona una solución más avanzada mediante firmas digitales y utiliza la infraestructura de claves públicas por omisión. Utiliza el estándar ISO9796 de validación/signatura digital que permite que la aplicación receptora establezca una prueba de que el mensaje procede de un emisor supuesto. Para proteger la confidencialidad del mensaje se utiliza el cifrador del atributo suministrado. El resumen SHA1 garantiza la integridad del mensaje y las funciones de cifrado/descifrado RSA aseguran que el mensaje sólo podrá restaurarlo el receptor al que va destinado. Al igual que con MQeMAttribute, se pueden utilizar todas las opciones de compresor y cifrador que ofrece MQSeries Everyplace. Para obtener una mejor optimización del tamaño, los certificados que se utilizan son minicertificados WTLS. La disponibilidad mutua de información necesaria para la autenticación (validación de firmas) y para el cifrado/descifrado se proporciona a través de la infraestructura por omisión de MQSeries Everyplace.

Un mensaje típico protegido por MQeMTrustAttribute tiene el formato:

```
RSA-enc{SymKey}, SymKey-enc {Data, DataDigest, DataSignature}
```

donde:

<b>RSA-enc:</b>	el cifrado RSA con la clave pública del emisor al que va destinado
<b>SymKey</b>	la clave simétrica pseudoaleatoria generada
<b>SymKey-enc</b>	cifrado simétrico con SymKey
<b>Data</b>	los datos del mensaje
<b>DataDigest</b>	el resumen de los datos del mensaje
<b>DigSignature</b>	la signatura digital del emisor de los datos del mensaje

La seguridad a nivel de mensajes es independiente de la seguridad a nivel de colas.

## seguridad

### El registro

El registro es el almacenamiento principal de la información relacionada con el gestor de colas y existe uno para cada gestor de colas. Cada gestor de colas utiliza el registro para guardar:

- Datos de configuración del gestor de colas
- Definiciones de colas
- Definiciones de colas remotas
- Definiciones del gestor de colas remotas
- Datos de usuario (incluida la información de seguridad que depende de la configuración)

Normalmente sólo el usuario legítimo del gestor de colas puede acceder al registro ya que está protegido con un PIN, pero hay una opción configurable que permite que los usuarios que estén más interesados en el espacio que en la seguridad puedan ignorar esta limitación.

### Entidades autenticables de MQSeries Everyplace

La seguridad basada en colas, que utiliza los minicertificados basados en la autenticación mutua, y la protección a nivel de mensajes, que utiliza la signatura digital, han creado un concepto de *entidad autenticable*. En el caso de la autenticación mutua es normal considerar la autenticación como algo que sucede entre dos usuarios (personas) pero, por lo general, en la mensajería no existe ningún concepto de usuario. Normalmente este concepto se gestiona en el nivel de la aplicación, esto es, por parte del usuario de los servicios de mensajería. MQSeries Everyplace, de forma deliberada, efectúa una abstracción del concepto de *destino de autenticación* y en lugar de *usuario* emplea el concepto de *entidad autenticable*. Este no excluye la posibilidad de que las entidades autenticables sean individuos, pero esta correlación la debería seleccionar la aplicación. Internamente, MQSeries Everyplace define todos los gestores de colas que pueden ser el origen o el destino de los servicios dependientes de minicertificados como una entidad autenticable. Además, MQSeries Everyplace también define las colas, que se han definido para utilizar autenticadores basados en minicertificados, como entidades autenticables. Por lo tanto, un gestor de colas que dé soporte a estos servicios puede tener una entidad autenticable, el gestor de colas, o un conjunto de entidades autenticables, el gestor de colas y todas las colas que utilicen un autenticador basado en certificados.

### El registro privado y las credenciales

Para que resulte práctica, cada entidad autenticable necesita sus propias credenciales. Esto supone dos retos. En primer lugar, cómo se ha de realizar el registro para obtener las credenciales y, en segundo lugar, dónde se pueden gestionar las credenciales de forma segura. Normalmente, estos retos son más difíciles de resolver que las técnicas criptográficas subyacentes. MQSeries Everyplace proporciona servicios por omisión que se pueden utilizar para que las entidades autenticables puedan efectuar un registro automático. Un registro privado (descendiente del registro base) y así habilitar la gestión segura de las credenciales privadas de la entidad autenticable, y un registro público (descendiente también del registro base) para gestionar el conjunto de credenciales públicas. El registro privado proporciona un registro base con muchas de las cualidades de un símbolo criptográfico o seguro, por

ejemplo, puede ser un depósito seguro de objetos públicos, como los minicertificados, y de objetos privados, como las claves privadas. Proporciona un mecanismo que permite que sólo el usuario autorizado acceda a los objetos privados. Da soporte a servicios, (por ejemplo, la signatura digital, el descifrado RSA), de tal modo que los objetos privados no abandonan nunca el registro privado. Proporciona una interfaz común, oculta el soporte de dispositivos subyacente, que actualmente está limitado al sistema de archivos local pero que en el futuro puede ampliarse perfectamente para poder correlacionar los símbolos transportables.

## Registro automático

MQSeries Everyplace proporciona servicios por omisión que dan soporte al registro automático. Estos servicios se activan automáticamente cuando se configura una entidad autenticable, por ejemplo, cuando se inicia un gestor de colas o cuando se define una nueva cola. En ambos casos, se activa el registro y se crean nuevas credenciales que se almacenan en el registro privado de la entidad autenticable. Los pasos para el registro automático incluyen la generación de un nuevo par de claves RSA, que protegen y guardan la clave privada en el registro privado y el empaquetado de la clave pública en una petición de *nuevo certificado* al servidor de minicertificados por omisión. Suponiendo que el servidor de minicertificados se haya configurado y esté disponible, devolverá el nuevo minicertificado de la entidad autenticable más su propio minicertificado, los cuales se almacenarán como sus nuevas credenciales, junto con la clave privada protegida, en el registro privado de la entidad autenticable. Aunque el registro automático proporciona un sencillo mecanismo para establecer las credenciales de la entidad autenticable, para la protección a nivel de mensajes (consulte MqeMTrustAttribute en las secciones anteriores), también es necesario el acceso a la clave pública (minicertificado) del receptor al que va destinado.

## Registro público y reproducción de certificados

MQSeries Everyplace proporciona servicios por omisión que permiten compartir las credenciales públicas (minicertificados) de las entidades autenticables entre los componentes de MQSeries Everyplace. Estos son un requisito previo para la seguridad a nivel de mensajes basada en MQeMTrust. El registro público de MQSeries Everyplace proporciona un depósito de minicertificados al que se puede acceder públicamente. Este depósito es similar al servicio de listín telefónico personal de un teléfono móvil, con la diferencia de que en lugar de números de teléfono almacena un conjunto de minicertificados de entidades autenticables con las que se contacta frecuentemente. El registro público no es un servicio totalmente pasivo. Si se accede al mismo para obtener un minicertificado que no contiene, y si se ha configurado con un componente de servidor inicial válido, el registro público intenta automáticamente obtener el minicertificado solicitado del registro público del servidor inicial. Estos servicios se pueden utilizar para obtener un servicio de reproducción de minicertificados automatizados e inteligentes que facilitan la disponibilidad del minicertificado adecuado en el momento oportuno.

## Utilización de los servicios de registro por parte de las aplicaciones

Aunque el gestor de colas MQSeries Everyplace se ha diseñado para poder beneficiarse de las ventajas de los servicios de registro privados y públicos, el acceso a estos servicios no está limitado. Es posible que las soluciones MQSeries Everyplace deseen definir y gestionar sus propias entidades autenticables, por ejemplo, los

## seguridad

usuarios. Entonces, se pueden utilizar los servicios del registro privado para realizar un registro automático y gestionar las credenciales de las nuevas entidades autenticables y los servicios del registro público para que las credenciales públicas estén disponibles cuando sean necesarias. Todas las entidades autenticables registradas se pueden utilizar como emisoras o receptoras de los servicios de protección a nivel de mensajes utilizando MQEMTrustAttribute.

### Servicio de emisión de minicertificado por omisión

MQSeries Everyplace proporciona un servicio de emisión de minicertificados por omisión que se puede configurar para satisfacer las peticiones de registro automático del registro privado. Con las herramientas que se proporcionan con MQSeries Everyplace, se puede configurar una solución y gestionar un servicio de emisión de minicertificados que emita minicertificados a un grupo de nombres de entidades bajo un estricto control. Las características de este servicio de emisión son:

- gestión del conjunto de entidades autenticables registradas
- emisión de minicertificado
- gestión de depósito de minicertificado WAP WTLS

Las herramientas que se proporcionan con MQSeries Everyplace permiten que un administrador de un servicio de emisión de minicertificados autorice a una entidad determinada a emitir minicertificados registrando su nombre de entidad y su dirección, y definiendo un PIN de petición de certificado que sólo se puede utilizar una vez. Normalmente, esto se lleva a cabo después de que se haya validado la autenticidad del solicitante mediante una comprobación fuera de línea. El PIN de petición de certificado se envía al usuario al que va destinado (por ejemplo, del mismo modo que los PIN de las tarjetas financieras se envían a los usuarios cuando se emite una nueva tarjeta). A continuación, se puede configurar el usuario del registro privado (por ejemplo, la aplicación MQSeries Everyplace o el gestor de colas MQSeries Everyplace) de modo que cuando se inicie la sesión proporcione este PIN de petición de certificado. Cuando el registro privado activa el registro automático, el servicio de emisión de minicertificados valida la petición de nuevo certificado resultante (dependiendo de si coinciden el nombre de la entidad y el PIN de petición de certificado presentados con los valores registrados previamente), emite el nuevo minicertificado y restaura el PIN de petición de certificado registrado para que no pueda volverse a utilizar. Todas las peticiones de nuevos minicertificados del registro automático se procesan en un canal seguro.

El conjunto de minicertificados que utiliza un servicio de emisión de minicertificados se conserva en el propio registro de este servicio de emisión. Cuando se vuelve a emitir un minicertificado (porque, por ejemplo, ha caducado), el minicertificado caducado se archiva.

### La interfaz de seguridad

Se proporciona una interfaz opcional que puede implementar un gestor de seguridad personalizado. Los métodos que presenta permiten al gestor de seguridad autorizar o rechazar las peticiones relacionadas con:

- La adición o supresión de alias de clases
- La definición de adaptadores

- La correlación de descriptores de archivos
- El proceso de mandatos de canal

---

## Personalización

### Normas

Las normas son clases Java que se utilizan para personalizar el comportamiento de MQSeries Everyplace cuando se producen diferentes modificaciones del estado. Las normas por omisión se proporcionan donde son necesarias, pero se pueden sustituir por normas específicas de la aplicación, o de la instalación, para adaptarlas a los requisitos del usuario. Los tipos de normas difieren en el modo en que se activan, no en la acción que pueden llevar a cabo. Las normas contienen lógica y, por lo tanto, realizan una amplia gama de funciones.

#### Normas de atributos

Se pasa el control a esta clase de norma cuando se intenta realizar un cambio de estado, por ejemplo, si se intenta modificar el:

- Autentificador
- Compresor
- Cifrador

Normalmente, la norma puede permitir o prohibir el cambio.

#### Normas de puente de MQSeries

Se pasa el control a esta clase de norma cuando el código entre MQSeries Everyplace y puente de MQSeries sufre un cambio de estado. Existe una clase de norma de puente diferente para determinar qué se ha de hacer en cada una de las situaciones siguientes:

- Cuando un escucha no puede entregar un mensaje a MQSeries Everyplace y éste procede de MQSeries. Debido, por ejemplo, a que el mensaje es demasiado grande o a que la cola no existe.
- El estado que se utiliza como puente de los objetos administrados debería iniciarse cuando se inicia la instancia del servidor.
- Cuando el puente encuentra que algo no va bien en la cola de sincronismo (Sync queue) en MQSeries (el almacenamiento permanente que se utiliza para la recuperación en caso de una anomalía global). La norma por omisión simplemente visualiza el problema.
- Cuando se ha de convertir un mensaje MQSeries Everyplace en un mensaje MQSeries, y viceversa. Los transformadores para efectuar la conversión entre mensajes MQSeries Everyplace y MQSeries no se derivan de ninguna clase MQeRule, sino que deben implementar la interfaz MQeTransformerInterface. Aparte de esto, los transformadores actúan como normas y se invocan cuando un mensaje necesita una conversión de formato.

### Normas de colas

Se pasa el control a esta clase de norma cuando la cola asociada ha sufrido un cambio de estado, por ejemplo:

- Se ha añadido un mensaje a una cola. Por ejemplo, para ver si se sobrepasa un umbral (de número de mensajes, de tamaño de mensajes, de prioridad no válida)
- Se han asignado o modificado las características de la cola
- Se ha abierto o se ha cerrado la cola
- Se ha de eliminar la cola

### Normas del gestor de colas

Se pasa el control a esta clase de norma cuando el gestor de colas sufre un cambio de estado, por ejemplo:

- Cuando se abre el gestor de colas. Por ejemplo, iniciar una hebra del temporizador subyacente que se ejecute para permitir las acciones de temporización.
- Se cierra el gestor de colas. Por ejemplo, finalizar la hebra del temporizador subyacente
- Se añade una nueva cola

---

## Tipos de conexiones

MQSeries Everyplace puede ofrecer soporte para operaciones de igual a igual y cliente-servidor. Un *cliente* puede iniciar la comunicación con un servidor. Un *servidor* sólo puede responder a las peticiones que inicia un cliente. En las operaciones *de igual a igual*, los dos elementos de la comunicación pueden iniciar los flujos en cualquier dirección. Estos tipos de conexiones requieren que estén disponibles y activos diferentes componentes de MQSeries Everyplace. Estos componentes son:

- **Escucha de canal:** que escucha las peticiones de conexión de entrada.
- **Gestor de canales:** que da soporte a varios conductos lógicos de comunicaciones simultáneos entre los extremos finales.
- **Gestor de colas:** que da soporte a las aplicaciones proporcionando posibilidades de mensajería y de gestión de colas.

En la Tabla 16 se muestra la relación entre estos componentes y los tipos de conexión. La conexión de tipo cliente-servidor describe aquella situación en la que MQSeries Everyplace puede operar tanto en modalidad de cliente como de servidor. La opción del servlet describe el caso en que se configura MQSeries Everyplace como un servlet HTTP en la que el servidor HTTP propiamente dicho es el responsable de estar a la escucha de las peticiones de conexión de entrada.

Tabla 16. Tipos de conexiones

	Gestor de colas	Gestor de canales	Escucha de canal
Cliente	Sí		
Cliente-servidor	Sí	Sí	Sí
De igual a igual	Sí		
Servidor	Sí	Sí	Sí

Tabla 16. Tipos de conexiones (continuación)

	Gestor de colas	Gestor de canales	Escucha de canal
Servlet	Sí	Sí	

Las aplicaciones MQSeries Everyplace desconocen directamente el tipo de conexión que utilizan los gestores de colas. Sin embargo, el tipo de conexión es importante ya que determina los recursos que tienen a su disposición las partes, qué gestores de colas pueden conectar con otros gestores de colas, el espacio de MQSeries Everyplace y qué conexiones pueden existir al mismo tiempo.

### Conexión de igual a igual

Un canal de igual a igual presenta las posibilidades de un gestor de canales y de un escucha de canal en un solo canal. Cuando se crea un canal de igual a igual entre dos gestores de colas, un gestor de colas debe actuar como escucha y el otro como iniciador de la conexión. En un tipo de conexión de igual a igual, un gestor de colas puede iniciar varias conexiones de igual a igual con otros gestores de colas; pero sólo puede responder a una petición de conexión de entrada y, a continuación, debe esperar a que se cierre este canal de igual a igual para poder responder a otra petición de este tipo. Los dos gestores de colas que participan en cualquier canal de igual a igual pueden iniciar acciones, de este modo, las aplicaciones de cada gestor de colas pueden, por ejemplo, acceder a las colas del otro.

Los canales de igual a igual no se pueden utilizar a través de cortafuegos ya que el destino de la petición de conexión de entrada puede no ser aceptado por el cortafuegos.

### Conexión cliente-servidor

Los canales estándar, que se utilizan para las conexiones de tipo cliente-servidor, no tienen posibilidades de escucha sino que dependen de un escucha independiente ubicado en el servidor y éste requiere un gestor de canales para poder manejar varios canales simultáneamente. El cliente inicia la petición de conexión y el servidor responde. Normalmente, un servidor puede manejar varias peticiones de entrada de los clientes. A través de un canal estándar, el cliente tiene acceso a los recursos del servidor. Si una aplicación del servidor necesita acceso síncrono a los recursos del cliente, será necesario un segundo canal en el que se inviertan las funciones. Sin embargo, dado que los canales propiamente dichos son bidireccionales, los mensajes dirigidos a un cliente desde la cola de transmisión de su servidor, se entregan a éste a través del canal estándar (cliente-servidor) que lo ha iniciado.

Un cliente puede ser un cliente para varios servidores a la vez. (Tenga en cuenta que no es necesario que el gestor de canales dé soporte a esta configuración porque los gestores de canales pueden manejar varios canales de entrada).

En general, el tipo de conexión cliente-servidor se puede utilizar a través de un cortafuegos ya que, normalmente, éste identifica como aceptable el destino de la conexión de entrada.

## tipos de conexiones

### Tipos de conexiones diferentes

Un solo gestor de colas puede iniciar las conexiones de igual a igual o de cliente-servidor y responder como servidor o como igual. En este caso, el escucha de canal de igual a igual y el escucha de canal estándar deben tener números de puertos diferentes.

---

## Clases

MQSeries Everyplace proporciona una selección de clases para determinadas funciones que permiten personalizar el comportamiento de MQSeries Everyplace según los requisitos específicos de las aplicaciones. En algunos casos, se documentan las interfaces con las clases, para poder desarrollar alternativas adicionales. La Tabla 17 resume las posibilidades. Se pueden identificar las clases de forma explícita o utilizando los nombres de alias.

*Tabla 17. Opciones de clases*

Clase	Alternativas suministradas	Interfaces documentadas
Administración	no	sí
Autenticadores	sí	no
Adaptador de comunicaciones	sí	sí
Tipo de comunicaciones	sí	no
Compresores	sí	no
Cifrador	sí	no
Anotaciones de sucesos	se proporciona un ejemplo	sí
Mensajes	no	sí
Almacenamiento en cola	sí	no
Normas	se proporcionan clases por omisión	sí
Rastreo	se proporcionan ejemplos	sí

## Carga de las aplicaciones

Cuando se configura un gestor de colas MQSeries Everyplace para que funcione como un cliente (o como un igual) la aplicación de inicio es la responsable de cargar cualquier otra aplicación en la JVM. Se pueden utilizar los recursos Java estándar para llevarlo a cabo, o el procedimiento de carga de clases que se incluye como parte de MQSeries Everyplace. De este modo, se pueden ejecutar varias aplicaciones en un solo gestor de colas dentro de la misma JVM. Alternativamente, se pueden utilizar varias JVM, pero cada una de ellas requerirá su propio gestor de colas y, cada una debe tener un nombre exclusivo.

Cuando se configura un gestor de colas MQSeries Everyplace como un servidor, MQSeries Everyplace es la aplicación de inicio. MQSeries Everyplace da soporte a una lista de clases de carga previa y estas clases se cargan por orden, antes de que el gestor de colas propiamente dicho se cargue.



---

## Capítulo 6. Redes MQSeries Everyplace y MQSeries

Aunque una red MQSeries Everyplace puede existir de forma autónoma, sin necesidad de un servidor o red MQSeries, en la práctica MQSeries Everyplace se utiliza normalmente como complemento de una instalación MQSeries existente. Este complemento permite ampliar su alcance a nuevas plataformas y dispositivos, y proporciona posibilidades avanzadas como, por ejemplo, la seguridad basada en colas o en mensajes o la mensajería síncrona. Desde la perspectiva de una aplicación MQSeries Everyplace, las colas y los gestores de colas MQSeries se pueden considerar simplemente como gestores de colas remotas y colas remotas adicionales. Sin embargo, existen diferentes limitaciones funcionales debido a que no se puede acceder directamente a las colas a través de los canales dinámicos de MQSeries Everyplace y de un gestor de colas MQSeries Everyplace, sino que se necesita que participe una pasarela MQSeries Everyplace. La pasarela puede enviar mensajes a varios gestores de colas MQSeries, tanto directa como indirectamente, a través de los canales de cliente MQSeries. Si la conexión es indirecta, los mensajes pasan a través de los canales de cliente MQSeries a un gestor de colas MQSeries intermediario y, a partir de ahí, avanzan por los canales de mensajes MQSeries hasta el gestor de colas de destino.

Los mensajes de una aplicación MQSeries destinados a MQSeries Everyplace se dirigen al gestor de colas MQSeries Everyplace y se ponen en cola de forma normal, con el direccionamiento de MQSeries definido (las definiciones del gestor de colas remotas) de tal modo que los mensajes de MQSeries Everyplace llegan a las colas de transmisión MQSeries específicas. Los canales MQSeries no se definen para las colas de transmisión, como se haría normalmente, sino que la pasarela MQSeries Everyplace extrae los mensajes de estas colas y asegura su entrega al destino de MQSeries Everyplace. Se puede configurar el número de colas de transmisión que se ha de utilizar (esto es, el número de rutas) y debe establecerse de modo que refleje el volumen de mensajes que se ha de entregar.

---

### Interfaz con MQSeries

La arquitectura de MQSeries Everyplace da soporte al concepto de uno o varios tipos de puentes opcionales entre MQSeries Everyplace y otros sistemas de mensajería.

En MQSeries Everyplace Versión 1.2 sólo se daba soporte a un puente de este tipo, el *puente MQSeries* que sirve de interfaz entre las redes MQSeries Everyplace y MQSeries. Este puente utiliza el cliente Java de MQSeries como interfaz entre uno o varios gestores de colas MQSeries y, de este modo, permite que los mensajes fluyan desde MQSeries Everyplace a MQSeries y viceversa. En la versión actual de MQSeries Everyplace, se recomienda uno de estos puentes por servidor, y que cada uno de estos esté asociado a varios *proxys de gestores de colas MQSeries* (definiciones de gestores de colas MQSeries). Será necesaria una definición de proxy de gestor de colas para cada gestor de colas MQSeries que se comunique con MQSeries Everyplace. Cada una de estas definiciones puede tener asociado uno o varios *servicios de conexión de cliente*, y cada uno de estos representará una conexión con un solo gestor de colas MQSeries. Estos servicios pueden utilizar cada uno una

## conexión a MQSeries

conexión de servidor MQSeries diferente con el gestor de colas y, opcionalmente, un conjunto diferente de propiedades como, por ejemplo, las rutinas de salida o puertas.

Una pasarela puede tener varios *escuchas* que utilicen dicha pasarela para conectarse al gestor de colas MQSeries y recuperar mensajes desde MQSeries para MQSeries Everyplace. Un escucha utiliza únicamente un servicio para establecer su conexión y cada escucha se conecta a una sola cola de transmisión del gestor de colas MQSeries. Cada escucha desplaza los mensajes desde una sola cola de transmisión MQSeries a cualquier ubicación de la red MQSeries Everyplace, a través del gestor de colas de pasarela principal. De este modo, un solo gestor de colas de pasarela puede canalizar los orígenes de los mensajes MQSeries en la red MQSeries Everyplace.

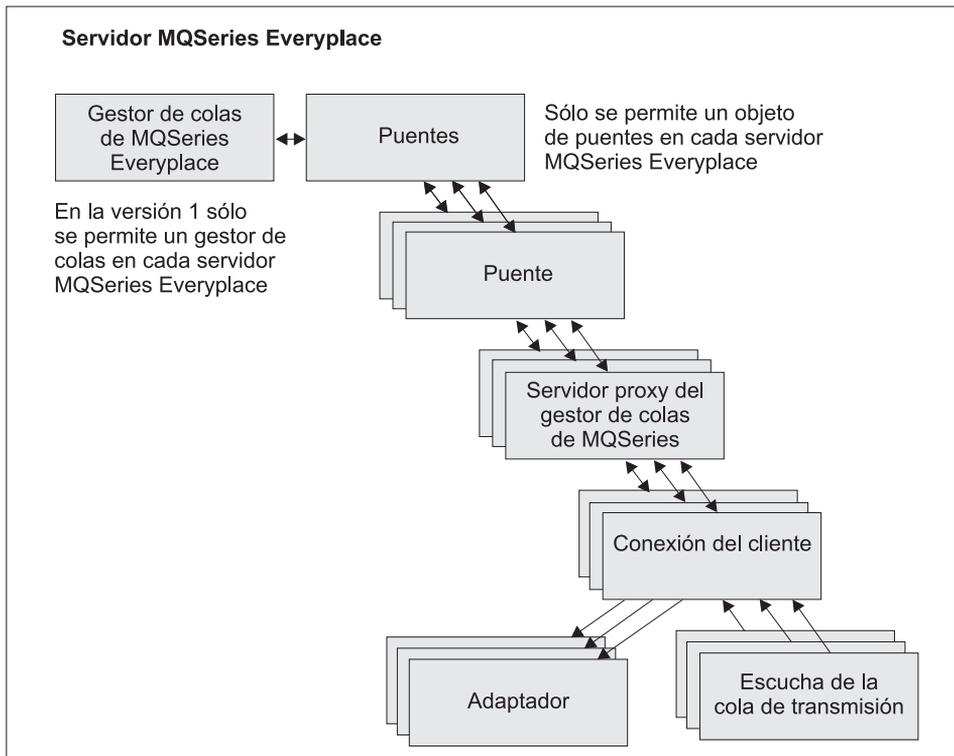


Figura 9. Jerarquía de los objetos del puente MQSeries

Cuando los mensajes se desplazan en la otra dirección, de MQSeries Everyplace a MQSeries, el gestor de colas de pasarela configura uno o varios objetos de *cola puente*. Cada objeto de cola puente se puede conectar directamente con cualquier gestor de colas y enviar sus mensajes a la cola de destino. De este modo, una pasarela puede asignar los mensajes MQSeries Everyplace dirigidos a través de un solo gestor de colas MQSeries Everyplace a cualquier gestor de colas MQSeries, tanto de forma directa como indirecta. El objeto de puente tiene las propiedades que se muestran en la Tabla 18 en la página 59.

Tabla 18. Propiedades del objeto de puente

Propiedad	Descripción
Nombre de puente	Lista de nombres de puente
Estado de ejecución	Estado: en ejecución o detenido

Los objetos de puente y los demás objetos de la pasarela se pueden iniciar y detener independientemente del gestor de colas MQSeries Everyplace. Si se inicia (o si se detiene) un objeto de pasarela de este tipo la acción también se aplica a todos sus elementos dependientes (todos los puentes, proxys del gestor de colas, conexiones de cliente y escuchas de cola de transmisión). La Tabla 19 muestra las propiedades del objeto de puente.

Tabla 19. Propiedades del puente

Propiedad	Descripción
Clase	Clase de puente
Transformador por omisión	La clase por omisión (la clase de norma) que se ha de utilizar para transformar un mensaje de MQSeries Everyplace en MQSeries (o viceversa) si no se ha asociado ninguna otra clase de transformador a la cola de destino
Intervalo de pulsación	La unidad de temporización básica que se utiliza para realizar las acciones en los objetos de puente
Nombre	El nombre del objeto de puente
Estado de ejecución	Estado: en ejecución o detenido
Clase de norma de inicio	La clase de norma que se utiliza cuando se inicia el objeto
Elemento dependiente del proxy del gestor de colas MQSeries	Lista de todos los proxys del gestor de colas que posee este puente

En los casos sencillos, se puede utilizar un transformador (norma) por omisión para manejar todas las conversiones de mensajes. Adicionalmente se puede establecer un transformador por escucha (para los mensajes de MQSeries a MQSeries Everyplace ) que altere temporalmente este valor por omisión. Para obtener un control más específico, se pueden establecer normas de transformación por cada cola de destino utilizando las definiciones de la cola puente de la pasarela. Esto se aplicaría tanto a las colas de destino de MQSeries Everyplace como de MQSeries.

El proxy del gestor de colas MQSeries contiene las propiedades específicas de un solo gestor de colas MQSeries. Las propiedades del proxy se muestran en la Tabla 20.

Tabla 20. Propiedades del proxy del gestor de colas MQSeries

Propiedad	Descripción
Clase	La clase del proxy del gestor de colas MQSeries

## conexión a MQSeries

*Tabla 20. Propiedades del proxy del gestor de colas MQSeries (continuación)*

Propiedad	Descripción
Nombre del sistema principal MQSeries	El nombre del sistema principal del IP que se ha utilizado para crear las conexiones con el gestor de colas MQSeries a través de las clases del cliente Java. Si no se especifica, se asume que el gestor de colas MQSeries está en la misma máquina que el puente y se utilizan los enlaces Java.
Nombre del proxy del gestor de colas MQSeries	El nombre del gestor de colas MQSeries
Nombre del puente propietario	El nombre del objeto de puente que posee este proxy del gestor de colas MQSeries
Estado de ejecución	Estado: en ejecución o detenido
Clase de norma de inicio	La clase de norma que se utiliza cuando se inicia el objeto del gestor de colas MQSeries
Elemento dependiente de la conexión de cliente	Lista de todos los proxys de la conexión de cliente que posee este fuente

La definición del servicio de conexión de pasarela contiene la información detallada que se necesita para realizar una conexión con el gestor de colas MQSeries. En la Tabla 21 se muestran las propiedades del servicio de conexión.

*Tabla 21. Propiedades del servicio de conexión del cliente*

Propiedad	Descripción
Clase de adaptador	La clase que se utilizará como adaptador de pasarela
CCSID*	El entero que representa el valor del CCSID de MQSeries que se ha de utilizar
Clase	La clase de servicio de la conexión de cliente del puente
Tiempo máximo de conexión desocupada	El tiempo máximo que se permite que una conexión permanezca desocupada antes de finalizarla
Contraseña MQSeries*	La contraseña que utiliza el cliente Java
Puerta MQSeries*	El número de la puerta IP que se ha utilizado para crear las conexiones con el gestor de colas MQSeries a través de las clases del cliente Java. Si no se especifica, se asume que el gestor de colas MQSeries está en la misma máquina que el puente y se utilizan los enlaces Java.
Clase de rutina de salida de recepción MQSeries*	Se utiliza para hacer coincidir la rutina de salida de recepción utilizada en el otro extremo del canal del cliente. La rutina de salida tiene asociada una serie de caracteres que permite pasar datos al código de salida
Clase de rutina de salida de seguridad MQSeries*	Se utiliza para hacer coincidir la rutina de salida de seguridad utilizada en el otro extremo del canal del cliente. La rutina de salida tiene asociada una serie de caracteres que permite pasar datos al código de salida

Tabla 21. Propiedades del servicio de conexión del cliente (continuación)

Propiedad	Descripción
Clase de rutina de salida de envío MQSeries*	Se utiliza para hacer coincidir la rutina de salida de envío utilizada en el otro extremo del canal del cliente. La rutina de salida tiene asociada una serie de caracteres que permite pasar datos al código de salida
ID de usuario de MQSeries*	El ID de usuario que utiliza el cliente Java
Nombre del servicio de conexión de cliente	El nombre del canal de conexión del servidor en la máquina MQSeries
Nombre del proxy del gestor de colas propietario	El nombre del proxy del gestor de colas propietario
Clase de norma de inicio	La clase de norma que se utiliza cuando se inicia el servicio de conexión de cliente del puente
Nombre de la cola de sincronismo	El nombre de la cola MQSeries que utiliza el puente para fines de sincronización
Clase de normas del depurador de colas de sincronismo	La clase de normas que se ha de utilizar cuando se encuentra un mensaje en la cola de sincronismo
Estado de ejecución	Estado: en ejecución o detenido
Nombre del puente propietario	El nombre del objeto de puente que posee esta conexión de cliente
Elemento dependiente de escucha XmitQ de MQ	Lista de todos los objetos escucha que utilizan esta conexión de cliente
*Se puede obtener información detallada sobre estos parámetros en la publicación <i>MQSeries Utilización de Java</i>	

La *clase de adaptador* se utiliza para enviar mensajes desde MQSeries Everyplace a MQSeries y la *cola de sincronismo* se utiliza para realizar un seguimiento del estado de este proceso. Su contenido se utiliza en situaciones de recuperación para garantizar una mensajería asegurada. Después de una conclusión normal, la cola está vacía. Pueden compartirla varias conexiones de cliente y varias definiciones de puente siempre que las rutinas de salida de recepción, envío y seguridad sean las mismas. Esta cola también se puede utilizar para almacenar el estado de los mensajes que se desplazan desde MQSeries a MQSeries Everyplace, dependiendo de las propiedades del escucha que esté utilizándose. La *clase de normas del depurador de colas de sincronismo* se utiliza cuando se encuentra un mensaje en la cola de sincronismo que indica que MQSeries Everyplace no ha confirmado un mensaje.

El tiempo máximo de conexión inactiva se utiliza para controlar la agrupación de conexiones de cliente Java que mantiene el servicio de conexiones de cliente del puente en su sistema MQSeries. Cuando una conexión MQSeries pasa a estar inactiva, debido a que no se utiliza, se inicia un temporizador y la conexión inactiva se descarta si el temporizador caduca antes de que vuelva a utilizarse la conexión. Crear conexiones MQSeries es una operación cara y este proceso garantiza que se vuelvan a utilizar de forma eficaz sin consumir demasiados recursos. Un valor cero indica que la agrupación de conexiones no debe utilizarse.

## conexión a MQSeries

El objeto de escucha, que desplaza los mensajes de MQSeries a MQSeries Everyplace, tiene las propiedades que se muestran en la Tabla 22.

Tabla 22. Propiedades del escucha

Propiedad	Descripción
Clase	La clase del escucha
Nombre de la cola de mensajes no entregados	La cola que se utiliza para contener los mensajes de MQSeries a MQSeries Everyplace que no se han podido entregar
Adaptador de almacenamiento de estados del escucha	El nombre de la clase del adaptador que se utiliza para almacenar la información sobre el estado
Nombre del escucha	El nombre de la cola XMIT de MQSeries que suministra los mensajes
Nombre del servicio de conexión de cliente propietario	Nombre del servicio de conexión de cliente
Estado de ejecución	Estado: en ejecución o detenido
Clase de norma de inicio	La clase de norma que se utiliza cuando se inicia el objeto de escucha
Clase de transformador	La clase de norma que se utiliza para determinar la conversión de un mensaje MQSeries para MQSeries Everyplace
Clase de norma de mensaje no entregado	La clase de norma que se utiliza para determinar la acción que se ha de realizar cuando no se pueden entregar mensajes de MQSeries a MQSeries Everyplace
Segundos de espera de mensaje	Una opción avanzada que se puede utilizar para controlar el rendimiento del escucha en circunstancias especiales

La *clase de norma de mensaje no entregado* determina qué acción se ha de llevar a cabo cuando no se puede entregar un mensaje de MQSeries a MQSeries Everyplace. Normalmente se coloca en la *cola de mensajes no entregados* del sistema MQSeries.

Para entregar mensajes de forma segura, la clase de escucha utiliza el *adaptador de almacenamiento de estados del escucha* para poder almacenar información sobre el estado, tanto en el sistema MQSeries Everyplace como en la cola de sincronismo del sistema MQSeries.

El escucha de colas de transmisión permite que las colas remotas MQSeries hagan referencia a las colas locales MQSeries Everyplace. También puede crear colas remotas MQSeries Everyplace que hagan referencia a colas locales MQSeries. Estas definiciones de colas remotas MQSeries Everyplace se denominan *colas puente de MQSeries* y se pueden utilizar para obtener, transferir y examinar mensajes de una cola MQSeries.

Una definición de cola puente de MQSeries puede contener los atributos siguientes:

Tabla 23. Propiedades de las colas puente de MQSeries

Propiedad	Descripción
Nombres de alias	Los nombres alternativos de la cola
Autentificador	Debe ser un valor nulo
Clase	La clase del objeto
Conexión de cliente	El nombre del servicio de conexión de cliente que se ha de utilizar
Compresor	Debe ser un valor nulo
Cifrador	Debe ser un valor nulo
Caducidad	Se pasa al transformador
Tamaño máximo de mensaje	Se pasa a la clase de normas
Modalidad	Debe ser síncrona
Proxy del gestor de colas MQ	El nombre del gestor de colas MQSeries al que debe enviarse el mensaje en primer lugar
Puente MQSeries	El nombre del puente que transmitirá el mensaje a MQSeries
Nombre	El nombre por el que se conoce a la cola MQSeries en MQSeries Everyplace
Gestor de colas propietario	El gestor de colas propietario de la definición
Prioridad	La prioridad que se utilizará con los mensajes (a menos que la altere temporalmente mediante un valor del mensaje)
Nombre de la cola MQSeries remota	El nombre de la cola MQSeries remota
Norma	La clase de norma que se utiliza para las operaciones de cola
Gestor de colas de destino	El gestor de colas MQSeries propietario de la cola
Transformador	El nombre de la clase de transformador que convierte el mensaje en formato MQSeries Everyplace al formato MQSeries
Tipo	Cola del puente MQSeries

**Nota:** Las clases del *cifrador*, del *autentificador* y del *compresor* definen un conjunto de atributos de cola que dictan el nivel de seguridad de cualquier mensaje que se pase a esta cola. Desde el momento en que el mensaje se envía inicialmente a MQSeries Everyplace hasta el momento en que se pasa a la cola puente de MQSeries, el mensaje está protegido con el nivel de seguridad de la cola como mínimo. Estos niveles de seguridad *no* son aplicables cuando la cola puente de MQSeries pasa el mensaje al sistema MQSeries, pues durante este tipo de transferencia se utilizan las rutinas de salida de seguridad, envío y recepción de la conexión de cliente. No se efectúa ninguna comprobación para asegurarse de que se mantiene el nivel de seguridad de la cola.

Las colas puente de MQSeries son sólo síncronas. Por este motivo, las aplicaciones asíncronas deben utilizar una combinación de colas del servidor inicial y colas para

## conexión a MQSeries

almacenar y reenviar MQSeries Everyplace, o definiciones de colas remotas asíncronas como paso intermedio al enviar mensajes a las colas puente de MQSeries.

Las aplicaciones utilizan las colas puente de MQSeries como cualquier otra cola remota MQSeries Everyplace, usando los métodos *putMessage*, *browseMessages* y *getMessage* de la clase *MQeQueueManager*. En estas invocaciones, el parámetro del nombre de cola es el nombre de la cola puente de MQSeries y el parámetro del nombre del gestor de colas es el nombre del gestor de colas MQSeries. Sin embargo, para que el servidor MQSeries Everyplace local acepte el nombre del gestor de colas, debe haber una definición de conexión con este nombre de gestor de colas MQSeries con el valor de nulo para todos los parámetros, incluido el nombre de canal.

**Nota:** existen algunas limitaciones en la utilización de *getMessage* y *browseMessages* con colas puente de MQSeries. No se pueden obtener o examinar mensajes de colas puente de MQSeries que señalan a definiciones de colas remotas MQSeries. Tampoco se pueden utilizar ID de configuración de no cero en las operaciones de obtener de cola puente de MQSeries. Esto significa que la operación *getMessage* en colas puente de MQSeries no ofrece la entrega asegurada. Si tiene que garantizar una operación *get*, debe utilizar los escuchas de cola de transmisión para transferir los mensajes de MQSeries.

La administración del puente de MQSeries se maneja del mismo modo que la administración de un gestor de colas MQSeries Everyplace normal, utilizando los mensajes de administración. En el objeto definido se definen nuevas clases de mensajes según corresponda. En la Tabla 13 en la página 37 se muestran las clases de mensajes de administración de la pasarela.

---

## Conversión de mensajes

Los mensajes MQSeries Everyplace destinados a MQSeries pasan a través del puente y se convierten a un formato MQSeries, utilizando un transformador por omisión o uno específico de la cola de destino. Un transformador personalizado permite una mayor flexibilidad, por ejemplo, se puede utilizar una subclase de la clase de objeto de mensaje de MQSeries Everyplace para que represente los mensajes de un tipo determinado a través de la red MQSeries Everyplace. En la pasarela, un transformador puede convertir el mensaje a un formato MQSeries utilizando el tipo de correlación entre campos y valores de MQSeries adecuados y también añadiendo datos específicos que representen el significado de la subclase.

El transformador por omisión de MQSeries Everyplace a MQSeries no puede beneficiarse de la información de la subclase pero se ha diseñado para que resulte práctico en situaciones muy diversas. Sus características son las siguientes:

- **Flujo de mensajes de MQSeries Everyplace a MQSeries:**

El transformador por omisión de MQSeries Everyplace a MQSeries funciona junto con la clase *MQeMQMsgObject*. Esta clase es una representación de todos los campos que se pueden encontrar en una cabecera de mensaje de MQSeries. Utilizando *MQeMQMsgObject*, la aplicación puede establecer valores (por ejemplo, la prioridad) utilizando métodos *set()*. De este modo, cuando se pasa *MQeMQMsgObject* (o un objeto derivado de la clase *MQeMQMsgObject*) a través

del transformador MQSeries Everyplace por omisión, éste (MQeBaseTransformer) obtiene los valores de MQeMSMsgObject y establece los valores correspondientes en el mensaje MQSeries (por ejemplo, el valor de prioridad se copia en el mensaje MQSeries).

Si el mensaje que se va a pasar no es un objeto MQeMQMsgObject, y si no se deriva de la clase MQeMQMsgObject, todo el mensaje MQSeries Everyplace se copia en el texto del mensaje MQSeries (*se canaliza*). El campo de formato de mensaje de la cabecera del mensaje MQSeries se establece de modo que indique que el mensaje MQSeries contiene un mensaje MQSeries Everyplace en formato "canalizado".

- **Flujo de mensajes de MQSeries a MQSeries Everyplace:**

Los mensajes MQSeries para MQSeries Everyplace se manejan de forma similar a aquellos que viajan en dirección contraria. El transformador por omisión inspecciona el campo de tipo de mensaje de la cabecera MQSeries y actúa según corresponda.

Si la cabecera MQSeries indica que hay un mensaje MQSeries Everyplace "canalizado", entonces se recompone el texto del mensaje MQSeries para que sea igual al mensaje MQSeries Everyplace original y, a continuación, se envía a la red MQSeries Everyplace.

Si el mensaje no es un mensaje MQSeries Everyplace "canalizado", entonces se extrae el contenido de la cabecera del mensaje MQSeries y se coloca en un objeto MQeMQMsgObject. El texto del mensaje MQSeries se trata como un sencillo campo de bytes y también se coloca en el objeto MQeMQMsgObject. A continuación, se envía MQeMQMsgObject a la red MQSeries Everyplace.

Esta clase MQeMQMsgObject y el comportamiento del transformador por omisión significan que:

- Un mensaje MQSeries Everyplace puede desplazarse desde una red MQSeries a una red MQSeries Everyplace sin modificaciones.
- Un mensaje MQSeries puede desplazarse desde una red MQSeries Everyplace a una red MQSeries sin modificaciones.
- Una aplicación puede MQSeries Everyplace puede dirigir cualquier aplicación MQSeries existente sin que se modifique la aplicación MQSeries.

---

## Función

Las colas remotas MQSeries pueden realizar de forma síncrona las operaciones de transferencia de mensajes MQSeries Everyplace desde un gestor de colas MQSeries Everyplace. Todas las demás operaciones de mensajería deben realizarse de forma asíncrona.

Los mensajes de administración de MQSeries Everyplace no se pueden enviar a un gestor de colas MQSeries. La cola de administración (AdminQ) no existe en este gestor de colas y el formato del mensaje de administración difiere del utilizado por MQSeries.

---

### Compatibilidad

Una red MQSeries Everyplace puede existir independientemente de MQSeries, pero en muchas situaciones se necesitan los dos productos juntos para satisfacer los requisitos de la aplicación. MQSeries Everyplace puede integrarse en una red MQSeries existente y, a continuación, se resume el alcance de la compatibilidad:

#### Direccionamiento y nombres:

- La semántica de direccionamiento es idéntica y se utiliza una dirección de gestor de colas/cola.
- Se utiliza de forma común el espacio de nombres ASCII

#### Aplicaciones:

MQSeries Everyplace puede dar soporte a aplicaciones MQSeries existentes sin necesidad de modificar la aplicación.

#### Canales:

Las pasarelas MQSeries Everyplace utilizan los canales de cliente MQSeries.

#### Intercambio de mensajes y contenido:

- intercambio de mensajes entre MQSeries Everyplace y MQSeries
- la red de mensajes no es visible (los mensajes desde MQSeries Everyplace o MQSeries pueden atravesar otras redes sin modificaciones)
- soporte mutuo para los campos identificados en la cabecera de mensaje MQSeries
- entrega de mensajes asegurada de una sola vez

MQSeries Everyplace Versión 1.2 no soporta todas las funciones de MQSeries. Además de las consideraciones de tipo de entornos, sistemas operativos y de comunicaciones, a continuación, se detallan algunas diferencias importantes. Sin embargo, tenga en cuenta que en MQSeries Everyplace se pueden realizar muchas tareas de aplicaciones de forma alternativa utilizando las funciones de MQSeries Everyplace o utilizando las ventajas que suponen las subclases, sustituyendo las clases suministradas o utilizando las normas, interfaces y otras características de personalización que incorpora el producto.

- No se da soporte a los clústers
- No se da soporte a la lista de distribución
- No existen mensajes agrupados ni segmentados
- No presenta posibilidades de equilibrio de carga/espera en caliente
- No existe un mensaje de referencia
- No hay opciones de informe
- No tiene soporte para colas compartidas
- No existe activación
- No hay soporte de unidades de trabajo, ni coordinación de tipo XA

Las características de rendimiento y posibilidad de crecimiento son diferentes.

---

## Entrega asegurada

Aunque tanto MQSeries Everyplace como MQSeries ofrece este tipo de entrega, cada uno lo proporciona de una manera diferente. Cuando un mensaje se desplaza desde MQSeries Everyplace a MQSeries, la transferencia del mensaje sólo se asegura si se utiliza la combinación de `putMessage` y `confirmPutMessage` (vea el apartado “Configuración del gestor de colas” en la página 31). Cuando un mensaje se desplaza desde MQSeries a MQSeries Everyplace, la transferencia únicamente se asegura si el mensaje MQSeries se ha definido como permanente.



---

## Capítulo 7. Interfaces de programación

*MQSeries Everyplace Systems Programming Interface (SPI)* es la interfaz de programación de MQSeries Everyplace. Se ofrece soporte para dos lenguajes, Java y C:

**La versión Java** proporciona acceso a todas las funciones de MQSeries Everyplace. En la publicación *MQSeries Everyplace for Multiplatforms Programming Reference* se proporciona información detallada sobre las clases, los métodos y los procedimientos, y, en la publicación *MQSeries Everyplace for Multiplatforms Guía de programación*, se proporcionan ejemplos de programación de MQSeries Everyplace .

**El soporte C para Palm** proporciona acceso a un subconjunto de funciones de MQSeries Everyplace para su utilización en dispositivos Palm. Puede obtener detalles sobre estas clases y procedimientos, además de instrucciones de programación, en la publicación *MQSeries Everyplace Native Client Information*



---

## Capítulo 8. Cómo empezar con MQSeries Everyplace

MQSeries Everyplace es una familia de productos que, en conjunto, proporcionan las herramientas necesarias para desarrollar, difundir y gestionar soluciones de gestión de colas y mensajería de MQSeries Everyplace. La familia consta de:

1. El *producto con licencia MQSeries Everyplace* (que se puede solicitar en soporte físico a IBM o se puede bajar de <http://software.boulder.ibm.com/dl/mqsem/mqsem-p>). El producto con licencia incluye:
  - Clases de Java MQSeries Everyplace
  - Clases de ayuda
  - Ejemplos de código fuente de aplicación
  - Programas de utilidad
  - Manuales de consulta
  - Información sobre licencia

El Producto de programa físico también incluye la autorización para utilizar el producto para usos que no sean de desarrollo (productivos) en determinadas plataformas. Para utilizarlo en máquinas mayores deben adquirirse unidades de mayor capacidad o con el puente de MQSeries.

2. Los SupportPacs de MQSeries Everyplace (que se pueden bajar de la web de <http://software.boulder.ibm.com/dl/mqsem/mqsem-p> (igual que la dirección anterior) o de <http://www.ibm.com/software/mqseries/everyplace>). Son complementos esenciales para el producto con licencia e incluyen, por ejemplo:

**EAP1: MQSeries Everyplace - Código de dispositivo para Palm OS**

Soporte de lenguaje de programación C para el desarrollo de aplicaciones MQSeries Everyplace Versión 1.0.1 en Palm OS. (Este código también se incluye en el CD del producto en el archivo eap1.zip)

**EP01: MQSeries Everyplace - Informe sobre rendimiento**

Analiza el rendimiento de MQSeries Everyplace en distintas plataformas de cliente

**ES01: MQSeries Everyplace - Herramienta de administración (MQExplorer v1.0)**

Herramienta genérica para todas las plataformas Java que habilita la administración gráfica fácil de gestores de colas MQSeries Everyplace

**ES02: MQSeries Everyplace - Explorer (MQ\_Explorer v1.2)**

Una herramienta de administración MQSeries Everyplace desarrollada exclusivamente para dar soporte al rango de sistemas operativos Windows de Microsoft

Las herramientas de gestión de los SupportPacs de MQSeries Everyplace representan un papel importante en todas las fases del desarrollo y la implantación de aplicaciones. Son más sofisticadas que los programas de utilidad que se incluyen con el producto con licencia y constituyen una ayuda esencial para la iniciación, la configuración, la inspección de redes en fase piloto y la gestión de sistemas de producción.

---

### Utilización de MQSeries Everyplace

Dada la gran variedad de usos de MQSeries Everyplace, el producto no se instala, configura y difunde del mismo modo que los demás miembros de la familia MQSeries. En este caso, el concepto fundamental es que, por lo general, la adopción de MQSeries Everyplace por parte de la empresa implica tres fases:

#### 1. Fase de desarrollo y creación de prototipos

En la primera fase de formación, desarrollo y creación de prototipos, el producto MQSeries Everyplace está disponible para la instalación y utilización de forma gratuita, sujeto a las condiciones de la licencia de desarrollo de IBM MQSeries Everyplace. Las aplicaciones MQSeries Everyplace se desarrollan utilizando las funciones de las rutinas C y las clases Java de MQSeries Everyplace. Estas aplicaciones se pueden empaquetar de diversos modos como, por ejemplo:

- Se puede establecer un gestor de colas MQSeries Everyplace como un daemon con una o más aplicaciones iniciadas en la Máquina Virtual Java y compartiendo un gestor de colas común.
- La aplicación incluye las clases de MQSeries Everyplace necesarias de modo que la aplicación se ejecute en máquinas en las que no se ha instalado MQSeries Everyplace, iniciando su propio gestor de colas en la máquina JVM.
- La aplicación utiliza las clases de MQSeries Everyplace que existen en la máquina de destino.

La licencia de desarrollo no incluye el soporte de IBM. Sin embargo, con la licencia de difusión sí se proporciona soporte durante y después del desarrollo de la aplicación (consulte más abajo).

#### 2. Fase de difusión

La fase de difusión representa la implantación y la utilización de las aplicaciones desarrolladas y, por este motivo, según los términos de la licencia de IBM MQSeries Everyplace, se necesitan unidades de capacidad para utilizar el producto. Las clases sólo se pueden distribuir con la aplicación con el consentimiento de IBM, o en caso de que los usuarios dispongan de la licencia necesaria para utilizarlas. De lo contrario, los usuarios deben proporcionar ellos mismos las clases necesarias.

#### 3. Fase de gestión

Posteriormente, cuando los gestores de colas MQSeries Everyplace están activos en la red, se necesitan herramientas para inspeccionarlos y gestionarlos. El soporte para MQSeries Everyplace se proporciona conforme a los términos del International Program License Agreement.

Este ciclo de vida de adopción justifica los distintos niveles de soporte con las plataformas. Para MQSeries Everyplace con unidades de capacidad (y SupportPacs de Categoría 3) IBM diferencia entre:

- Plataformas en las que se ofrece soporte para la instalación y el desarrollo de aplicaciones:

- se aceptan informes de problemas durante la instalación o el desarrollo de aplicaciones y la utilización
- Plataformas en las que se permite la difusión de aplicaciones, pero para las que no se ofrece soporte directamente:
  - es posible que se necesiten los informes de problemas para reproducirlos en una plataforma para la que se ofrezca soporte
- Plataformas en las que se ofrece soporte para la difusión de aplicaciones:
  - se aceptan informes de problemas resultado de la difusión de aplicaciones

---

## Cómo adquirir experiencia

Existen muchos procedimientos para empezar a utilizar MQSeries Everyplace. La experiencia sugiere que activar y ejecutar un gestor de colas y, a continuación, una red MQSeries Everyplace simple, es una forma productiva de familiarizarse con el producto y sus conceptos. Después, la creación de una aplicación sencilla puede ser una preparación lógica para un estudio con detenimiento de los detalles del producto. En las primeras etapas, por lo general, no tiene ninguna utilidad examinar otros miembros de la familia MQSeries pero, más adelante, cuando lo que interesa son las funciones de puente, esta información resulta esencial.

Desde la perspectiva de esta estrategia, se recomienda a los nuevos usuarios que adopten la propuesta siguiente:

Deden comprender los conceptos más importantes que se presentan en la publicación *MQSeries Everyplace for Multiplatforms Introducción* (este manual) y, a continuación, llevar a cabo lo siguiente:

1. Si disponen del acceso necesario a una máquina en la que se ejecute un sistema operativo Windows de Microsoft, deben bajar MQe\_Explorer, SupportPac ES02 (MQe\_Explorer v 1.2). No es necesario instalar previamente el producto con licencia, pero si no se hace, el usuario está limitado al uso de desarrollo bajo los términos de la licencia.
2. Deben seguir las instrucciones específicas que se indican a continuación, en el apartado “Primera utilización de ES02: MQe\_Explorer”.

### Primera utilización de ES02: MQe\_Explorer

MQe\_Explorer incluye una Guía del usuario completa que describe procedimientos rápidos para configurar el primer gestor de colas. Por lo general, este manual se recomienda a un público más amplio que únicamente a los administradores de MQe\_Explorer, puesto que incluye varios scripts de ejemplo que ilustran ejemplos importantes de las operaciones de MQSeries Everyplace. El SupportPac incluye dos versiones ejecutables de MQe\_Explorer:

#### MQe\_ExplorerX.exe

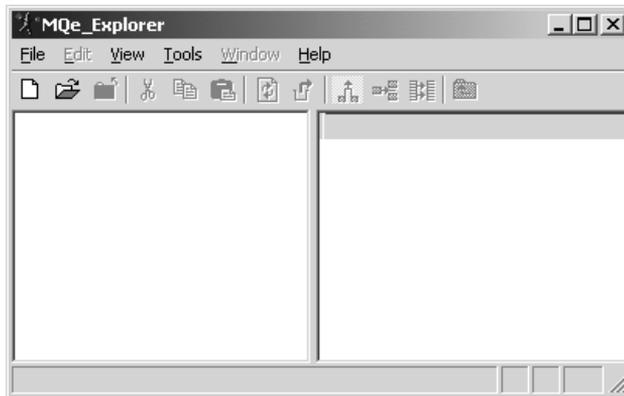
Esta versión incluye dentro del archivo .exe todas las clases de MQSeries Everyplace que se necesitan para su utilización. MQe\_ExplorerX.exe se ejecuta sin que se haya instalado MQSeries Everyplace en la máquina. Resulta ideal para los usuarios que están en la fase de introducción.

### MQe\_Explorer.exe

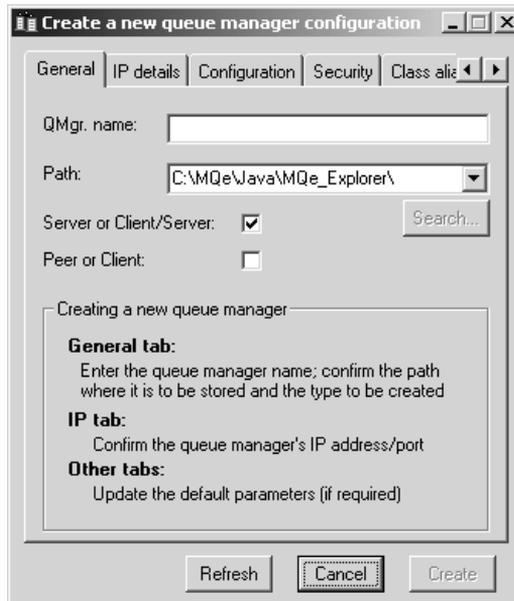
Esta versión depende de si se ha instalado MQSeries Everyplace previamente. Las ventajas son que incluye el último nivel de las bibliotecas MQSeries Everyplace y que ocupa mucho menos espacio. Está destinada a los desarrolladores y los administradores.

Como ejemplo de lo fácil que resulta crear gestores de colas, las instrucciones resumidas siguientes muestran la potencia de MQSeries Everyplace.

1. Efectúe una doble pulsación sobre el icono MQe\_ExplorerX.exe  .  
Un mensaje indica que no se ha encontrado ninguna opción guardada. Pulse **Aceptar** (este mensaje no se vuelve a mostrar). Se muestra la ventana de administración siguiente:



2. Pulse el icono nuevo  de la barra de herramientas.  
Se crea un nuevo gestor de colas. Se muestra la ventana siguiente:



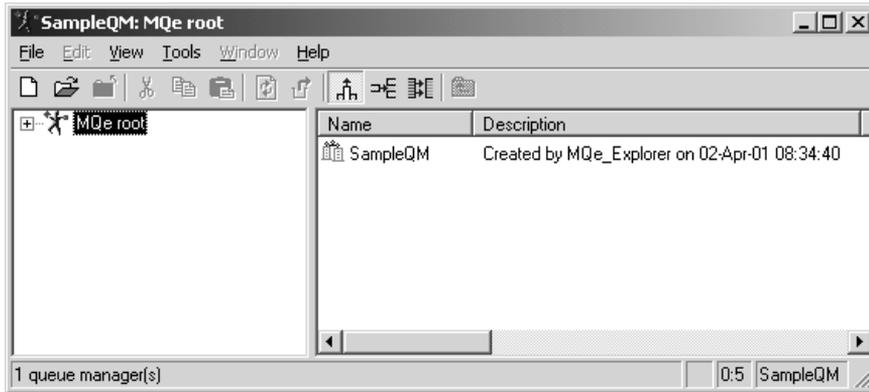
3.
  - a. Escriba el nombre de un gestor de colas (por ejemplo, SampleQM).
  - b. Seleccione el separador **IP details (Detalles de IP)**



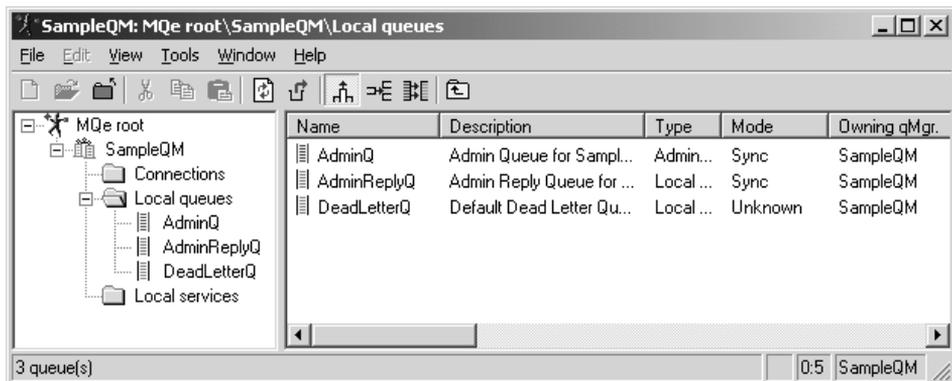
- 4.

## cómo empezar

- a. Escriba una dirección IP. La dirección es la dirección IP de la máquina que se está utilizando pero, en esta etapa, se puede utilizar el valor 127.0.0.1 (sistema principal local).
- b. Pulse el botón en **Crear**. Se emite un mensaje que identifica un archivo de inicialización que se ha creado (necesita el nombre para poder acceder posteriormente a este gestor de colas).
- c. Pulse en **Aceptar**. Se muestra la ventana siguiente:



5. Se ha creado un gestor de colas de servidor, que ejecuta su propia JVM. Está a la escucha de las peticiones de conexión de canal cliente/servidor de entrada en la puerta 8081. Si se expanden todos los símbolos + del árbol, situado en el panel de la izquierda y se cambia el tamaño de la ventana y de los paneles, se puede observar que se han creado cuatro colas.



6. Para realizar más pruebas, siga las instrucciones que se indican en la publicación *MQe\_Explorer User Guide*. Puede crear colas, conexiones, mensajes e incluso redes MQSeries Everyplace completas. Las aplicaciones se pueden cargar en este mismo gestor de colas y se pueden ejecutar simultáneamente con MQe\_Explorer.

---

## Apéndice. Avisos

Esta información se ha desarrollado para productos y servicios que se ofrecen en los Estados Unidos. Es posible que IBM no ofrezca en otros países los productos, servicios o características de los que trata esta publicación. Consulte al representante de IBM para obtener información sobre los productos y servicios que están disponibles en su área. Cualquier referencia a un producto, programa o servicio de IBM no pretende implicar que únicamente pueda utilizarse dicho producto, programa o servicio de IBM. En su lugar, puede utilizarse cualquier producto, programa o servicio que no infrinja los derechos de propiedad intelectual de IBM. Sin embargo, queda bajo la responsabilidad del usuario el evaluar y comprobar su funcionamiento con cualquier producto, programa o servicio que no sea de IBM.

IBM puede tener patentes o aplicaciones pendientes de patentes que traten el tema descrito en este documento. La adquisición de este documento no le otorga ninguna licencia sobre estas patentes. Puede enviar sus consultas, por escrito, a:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
EE.UU.

**El siguiente párrafo no se aplica al Reino Unido ni a ningún otro país en el que dichas medidas no sean coherentes con la normativa local:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROPORCIONA ESTA PUBLICACIÓN "TAL CUAL" SIN GARANTÍA DE NINGÚN TIPO, YA SEA EXPLÍCITA O IMPLÍCITA, PERO SIN LIMITARSE A LAS GARANTÍAS IMPLÍCITAS DE NO VULNERACIÓN, COMERCIALIZACIÓN O IDONEIDAD PARA UNA FINALIDAD DETERMINADA. Algunos estados no permiten la renuncia a las garantías explícitas o implícitas en determinadas transacciones, por lo tanto, es posible que esta declaración no sea aplicable en su caso.

Esta información puede contener imprecisiones técnicas o errores tipográficos. Periódicamente se efectúan cambios en la información que contiene esta publicación. Estos cambios se incorporarán en las nuevas ediciones. En cualquier momento, IBM puede realizar mejoras y/o cambios en el(los) producto(s) y/o programa(s) descrito(s) en esta publicación sin aviso previo.

Cualquier referencia a sitios web que no sean de IBM se proporciona únicamente para su comodidad y de ningún modo son una recomendación de dichos sitios web. Los materiales de estos sitios web no forman parte de los materiales de este producto IBM y su utilización será de su responsabilidad.

IBM puede utilizar o distribuir la información que le suministre en cualquier modo que crea conveniente sin incurrir por ello con ninguna obligación para con usted.

## avisos

Los usuarios bajo licencia de este programa que deseen intercambiar información a fin de: (i) habilitar el intercambio de información entre programas creados independientemente y otros programas (incluido este) y (ii) el uso mutuo de la información intercambiada, deberán ponerse en contacto con:

IBM United Kingdom Laboratories,  
Mail Point 151,  
Hursley Park,  
Winchester,  
Hampshire  
Reino Unido  
SO21 2JN

Dicha información puede estar disponible según los términos y condiciones correspondientes, incluido en algunos casos el abono de una cuota.

IBM proporciona el programa bajo licencia que describe esta información y todo el material bajo licencia disponible para el mismo según los términos del acuerdo con el cliente de IBM, (IBM Customer Agreement), del acuerdo de programas bajo licencia internacional de IBM, (IBM International Program License Agreement), o de cualquier acuerdo equivalente entre las partes.

---

## Marcas registradas

Los siguientes términos son marcas registradas de International Business Machines Corporation en los Estados Unidos, en otros países o en ambos.

AIX    AS/400    IBM    MQSeries    OS/390

Microsoft, Windows, Windows NT y el logotipo de Windows son marcas registradas de Microsoft Corporation en los Estados Unidos o en otros países.

Java y todos los logotipos y marcas registradas basadas en Java son marcas registradas de Sun Microsystems, Inc. en los Estados Unidos o en otros países.

Otros nombres de empresas, productos y servicios pueden ser marcas registradas o marcas de servicio de terceros.

---

## Glosario

Este glosario describe los términos que se utilizan en esta publicación y las palabras que se utilizan de un modo distinto a su significado cotidiano. En algunos casos, es posible que una definición no sea la única aplicable a un término, pero proporciona el sentido determinado con que esta palabra se utiliza en esta publicación.

Si no encuentra el término que está buscando, consulte el índice o el diccionario *IBM Dictionary of Computing*, New York: McGraw-Hill, 1994.

**autenticador.** Un programa que comprueba y verifica los emisores y receptores de los mensajes.

**canal.** Consulte *canal dinámico* y *canal MQI*.

**canal dinámico.** Un canal dinámico conecta los dispositivos MQSeries Everyplace y transfiere mensajes síncronos y asíncronos y respuestas de forma bidireccional.

**canal MQI.** Un canal MQI conecta un cliente MQSeries con un gestor de colas de un sistema servidor y transfiere las llamadas MQI y respuestas de forma bidireccional.

**cifrador.** Un programa que cifra un mensaje para generar seguridad durante la transmisión.

**clase.** Una clase es una agrupación encapsulada de datos y de métodos para operar en los datos. Se puede crear una instancia de una clase para generar un objeto que será una instancia de dicha clase.

**cliente.** En MQSeries, un cliente es un componente de ejecución que proporciona a las aplicaciones de usuarios locales el acceso a los servicios de gestión de colas de un servidor.

**cola.** Una cola es un objeto MQSeries. Las aplicaciones de colas de mensajes pueden poner mensajes en cola y obtener mensajes de una cola.

**cola de mensajes.** Consulte cola

**compresor.** Un programa que comprime un mensaje para disminuir el volumen de los datos que se han de transmitir.

**encapsulación.** La encapsulación es una técnica de programación orientada a objetos que convierte en privados o protegidos los datos de un objeto, y permite a los programadores acceder y manipular los datos únicamente a través de las llamadas a método.

**gestión de colas de mensajes.** Una técnica de programación en la que cada programa, dentro de una aplicación, se comunica con los demás programas colocando mensajes en las colas.

**gestor de canales.** Un objeto MQSeries Everyplace que da soporte a varios conductos lógicos de comunicaciones simultáneos entre los puntos de terminación.

**gestor de colas.** Un gestor de colas es un programa del sistema que proporciona servicios de gestión de colas de mensajes para las aplicaciones.

**HTML (Hypertext Markup Language).** Un lenguaje que se utiliza para definir información que se ha de visualizar en la World Wide Web.

**instancia.** Una instancia es un objeto. Cuando se crea una instancia de una clase para generar un objeto, se dice que este objeto es una instancia de la clase.

**interfaz.** Una interfaz es una clase que contiene únicamente métodos abstractos y no variables de instancias. Una interfaz proporciona un conjunto común de métodos que se puede implementar mediante subclases de diferentes clases.

**Interfaz de programas de aplicación (API).** Una interfaz de programas de aplicación consta de las funciones y variables que los programadores pueden utilizar en sus aplicaciones.

**Internet.** Internet es una red pública cooperativa de información compartida. Físicamente, Internet

utiliza un subconjunto de los recursos totales de todas las redes de telecomunicaciones públicas existentes. Técnicamente, lo que diferencia a Internet como red pública cooperativa es el uso de un conjunto de protocolos denominado TCP/IP (Protocolo de control de transmisión/Protocolo Internet).

**Java Developers Kit (JDK).** Un paquete de software distribuido por Sun Microsystems para programadores de Java. Incluye el intérprete de Java, las clases Java y las herramientas para el desarrollo de Java: el compilador, depurador, desensamblador, visor de applets, generador de archivos de apéndices y generador de documentación.

**Java Naming and Directory Service (JNDI).** Una API especificada en el lenguaje de programación Java. Proporciona funciones de denominación y directorio para las aplicaciones escritas en el lenguaje de programación Java.

**Lightweight Directory Access Protocol (LDAP).** LDAP es un protocolo cliente/servidor para acceder al servicio de directorios.

**mensaje.** En las aplicaciones de gestión de colas de mensajes, un mensaje es una comunicación que se envía entre programas.

**mensajería asíncrona.** Un método de comunicación entre programas en el que éstos colocan mensajes en colas de mensajes. Con la mensajería asíncrona, el programa emisor continúa con su propio proceso sin tener que esperar la respuesta al mensaje. Compare con *mensajería síncrona*.

**mensajería síncrona.** Un método de comunicación entre programas en el que éstos colocan mensajes en colas de mensajes. Con la mensajería síncrona, el programa emisor espera una respuesta a su mensaje antes de reanudar su propio proceso. Compare con *mensajería asíncrona*.

**método.** El método es el término de la programación orientada a objetos con que se designa una función o procedimiento.

**MQSeries.** MQSeries es una familia de programas bajo licencia de IBM que proporcionan servicios de gestión de colas de mensajes.

**navegador web.** Un programa que da formato y visualiza información que se distribuye en la World Wide Web.

**objeto.** (1) En Java, un objeto es una instancia de una clase. Una clase da forma a un grupo de cosas y un objeto da forma a un miembro determinado de dicho grupo. (2) En MQSeries, un objeto es un gestor de colas, una cola o un canal.

**paquete.** En Java, un paquete es una forma de proporcionar a una parte del código Java acceso a un conjunto de clases específicos. El código Java forma parte de un paquete determinado que tiene acceso a todas las clases del paquete y a todos los métodos y campos no privados de las clases.

**pasarela.** Una pasarela MQSeries Everyplace (o servidor) es un sistema que ejecuta el código MQSeries Everyplace, incluido un gestor de canales.

**Personal Digital Assistant (PDA).** Un ordenador personal de tamaño bolsillo.

**privado.** Un campo privado no puede visualizarse fuera de su propia clase.

**protegido.** Un campo protegido sólo puede visualizarse dentro de su propia clase, dentro de una subclase, o dentro de los paquetes de los que forma parte la clase

**pública.** Una clase o interfaz pública puede verse en cualquier lugar. Una variable o método público se puede visualizar dondequiera que se pueda visualizar su clase

**puente.** Un objeto MQSeries Everyplace que permite que los mensajes fluyan entre MQSeries Everyplace y otros sistemas de mensajería, incluido MQSeries.

**servidor.** (1) Un servidor MQSeries Everyplace es un dispositivo que tiene configurado un gestor de canales MQSeries Everyplace. (2) Un servidor MQSeries es un gestor de colas que proporciona

servicios de gestión de colas de mensajes a las aplicaciones de clientes que se ejecutan en una estación de trabajo remota. (3) De forma más general, un servidor es un programa que responde a las peticiones de información en el modelo de flujo de información particular entre dos programas de tipo cliente/servidor. (3) El sistema en el que se ejecuta un programa servidor.

**servlet.** Un programa Java diseñado para ejecutarse únicamente en un servidor web.

**subclase.** Una subclases es una clase que amplía otra. La subclase hereda las variables y métodos públicos y protegidos de su superclase.

**superclase.** Una superclase es una clase que se amplía mediante alguna otra clase. Las variables y métodos públicos y protegidos de la superclase están a disposición de la subclase.

**TCP/IP (Transmission Control Protocol/Internet Protocol).** Un conjunto de protocolos de comunicación que da soporte a funciones de conectividad de igual a igual tanto en redes de área local como en redes de área amplia.

**Web.** Vea World Wide Web.

**World Wide Web (Web).** La World Wide Web es un servicio de Internet, basado en un conjunto común de protocolos, que permite que un sistema servidor configurado de una manera determinada pueda distribuir documentos a través de Internet de una forma estándar.



---

## Bibliografía

Publicaciones relacionadas:

- *MQSeries Everyplace for Multiplatforms Léame primero*, SC10-3547
- *MQSeries Everyplace for Multiplatforms Programming Reference*, SC34-5846
- *MQSeries Everyplace for Multiplatforms Guía de programación*, SC10-3548
- *MQSeries Everyplace for Multiplatforms Native Client Information*, GC34-5883
- *MQSeries An Introduction to Messaging and Queuing*, GC33-0805-01
- *MQSeries para Windows NT Versión 5.1 Comienzo rápido*, GC10-3341-00



# Índice

## A

a quién va dirigido este manual v  
acciones de rastreo en MQSeries Everyplace 42  
acerca de este manual v  
adaptadores MQSeries Everyplace 41  
administración con MQSeries Everyplace 37  
anotaciones de sucesos 42  
aplicaciones, carga de 55  
aplicaciones MQSeries Everyplace 11  
asegurada, entrega de mensajes 67  
asíncrona, mensajería 42  
autenticables, entidades 48  
avisos legales 77

## B

basada en colas, seguridad 45

## C

canales 7  
canales, de cliente 15  
canales, dinámicos 15, 40  
canales cliente-servidor 7  
canales de cliente 15  
canales de igual a igual 7  
canales dinámicos 7, 15, 40  
carga de las aplicaciones 55  
certificados, reproducción 49  
cifrado 43  
clases de MQSeries Everyplace 54  
cliente MQSeries 6  
colas del servidor inicial 23  
colas locales 21  
colas MQSeries Everyplace 21  
colas para almacenar y reenviar 23  
colas puente de MQSeries 24  
colas remotas 22  
compatibilidad con MQSeries 66  
compresión 43  
comunicaciones 52  
conceptos del producto 15  
conexión, cliente-servidor 53  
conexión cliente-servidor 53  
conexión de igual a igual 53  
configuración 51  
configuraciones, ejemplo 8  
configuraciones de ejemplo 8  
conocimientos previos necesarios v  
conversión de mensajes 64

## D

de igual a igual. conexión 53  
descripción 1  
diferentes, tipos de conexiones 54

Dispositivos, MQSeries Everyplace 15  
dispositivos MQSeries Everyplace 15

## E

entidades autenticables 48  
entornos, software 3  
entornos de software 3  
entrega asegurada de mensajes 67  
escucha de canal 52

## F

formato de los datos de las operaciones de vuelco 21

## G

gestión de conexiones de marcación 41  
gestor de canales 52  
gestor de colas 52  
gestores de colas 6, 7  
gestores de colas, objetos 34  
gestores de colas MQSeries Everyplace 27

## I

interfaces de programación 69  
interfaz con MQSeries 57  
interfaz de seguridad 50

## L

la familia MQSeries 5  
lectores v  
legales, avisos 77  
local, seguridad 45  
locales, colas 21

## M

marcas registradas 78  
mensaje, objetos 16  
mensajería asíncrona 42  
mensajería de sistema principal vi, 5  
mensajería distribuida vi, 5  
mensajería general vi, 5  
mensajería MQSeries 5  
mensajería síncrona 43  
mensajes, de administración 37  
mensajes de administración 37  
minicertificados 48  
MQeAttribute 47  
MQeMTrustAttribute 47  
MQSeries, cliente 6  
MQSeries, compatibilidad con 66  
MQSeries, interfaz con 57  
MQSeries, mensajería 5  
MQSeries, servidor 6  
MQSeries Everyplace, adaptadores 41

- MQSeries Everyplace, aplicaciones 11
- MQSeries Everyplace, clases 54
- MQSeries Everyplace, colas 21
- MQSeries Everyplace, gestores de colas 27
- MQSeries Everyplace, normas 51
- MQSeries Everyplace, objetos 16
- MQSeries Everyplace, redes 42, 57
- MQSeries Everyplace, registro 15
- MQSeries Everyplace, seguridad 43
- MQSeries Everyplace, servidor inicial 23
- MQSeries Integrator vi, 5
- MQSeries Workflow vi, 5

## N

- nivel de mensajes, seguridad 46
- normas de atributos 51
- normas de colas 52
- normas de MQSeries Everyplace 51
- normas de puente de MQSeries 51
- normas del gestor de colas 52

## O

- objeto de escucha 62, 63
- objeto de puente 58, 59
- objeto del proxy del gestor de colas 59
- objetos de mensaje 16
- objetos de MQSeries Everyplace 16
- operaciones de los gestores de colas 34

## P

- para almacenar y reenviar, colas 23
- pasarelas, MQSeries Everyplace 15
- pasarelas MQSeries Everyplace 15
- personalización 51
- posibilidades 11
- privado, registro 48
- producto, conceptos 15
- programación, interfaces 69
- público, registro 49
- puente, MQSeries 57
- puente de MQSeries 7, 57
- puente de MQSeries, colas 24

## R

- redes MQSeries 57
- redes MQSeries Everyplace 42, 57
- registro 48
- registro automático 49
- registro de MQSeries Everyplace 15, 48
- registro privado 48
- registro público 49
- remotas, colas 22
- reproducción de certificados 49
- requisitos de los usuarios 12
- requisitos previos 3

## S

- seguridad, interfaz 50
- seguridad a nivel de mensajes 46
- seguridad basada en colas 45
- seguridad en MQSeries Everyplace 43
- seguridad local 45
- servicio de emisión de minicertificados 50
- servidor MQSeries 6
- síncrona, mensajería 43
- sistemas operativos necesarios 3
- sistemas operativos soportados 3
- soportados, sistemas operativos 3
- SPI 69
- supervisión 39

## T

- términos v
- tipos de conexiones 52
- tipos de conexiones diferentes 54
- transformadores 64

## U

- usuarios, requisitos 12

## V

- visión general 1

---

## Envío de comentarios a IBM

Si considera algún aspecto de este libro particularmente bueno o malo, utilice uno de los métodos que se describen a continuación para enviar sus comentarios a IBM.

No tenga reparos en comentar aspectos que considere errores u omisiones específicos, así como sobre su precisión, organización, tema o integridad del libro.

Rogamos limite sus comentarios a la información contenida en este libro y a la forma en que está presentada.

**Para realizar comentarios sobre las funciones de los productos o sistemas IBM, póngase en contacto con su representante de IBM o su distribuidor IBM autorizado.**

Cuando envíe comentarios a IBM, está concediendo a IBM un derecho no exclusivo para el uso o distribución de sus comentarios de la manera que considere apropiada, sin incurrir en obligación alguna con Ud.

Puede enviar sus comentarios a IBM de cualquiera de las formas siguientes:

- Por correo, a esta dirección:

User Technologies Department (MP095)  
IBM United Kingdom Laboratories  
Hursley Park  
WINCHESTER,  
Hampshire  
SO21 2JN  
Reino Unido

- Por fax:
  - Desde fuera del Reino Unido, debe marcar su código de acceso internacional seguido de 44–1962–842327
  - Desde el Reino Unido, marque 01962–842327
- De forma electrónica, utilice el ID de red adecuado:
  - IBM Mail Exchange: GBIBM2Q9 at IBMMAIL
  - IBMLink: HURSLEY(IDRCF)
  - Internet: idrcf@hursley.ibm.com

Cualquiera que sea el método que utilice, asegúrese de incluir la información siguiente:

- El título y el número de pedido de la publicación
- El tema al que se refieren sus comentarios
- Su nombre y dirección/número de teléfono/número de fax/ID de red.







Printed in Denmark by IBM Danmark A/S

GC10-3546-02

