

MQSeries<sup>®</sup> Everyplace



# Introducción

*Versión 1*



MQSeries<sup>®</sup> Everyplace



# Introducción

*Versión 1*

## **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 63

### **Aviso sobre licencia**

MQSeries Everyplace Versión 1 es un kit de herramientas que permite a los usuarios escribir aplicaciones MQSeries Everyplace y crear un entorno en el que ejecutarlas.

Las condiciones de la licencia de compra de este kit de herramientas determinan el entorno en el que puede utilizarse:

*Si MQSeries Everyplace se adquiere para utilizarlo como un **dispositivo** (cliente) **no** se podrá utilizar para crear un **gestor de canales MQSeries Everyplace** ni un **escucha de canal MQSeries Everyplace**.*

*La presencia de un **gestor de canales MQSeries Everyplace** o de un **escucha de canal MQSeries Everyplace** define un entorno de **pasarela** (servidor), para lo que es necesaria una licencia de pasarela.*

### **Primera edición (Junio 2000)**

Este manual es la traducción del original inglés "MQSeries® Everyplace Introduction Version 1", (GC34-5843-00).

Esta edición se aplica a MQSeries Everyplace Versión 1.0 y a todos los releases y modificaciones posteriores hasta que se indique lo contrario en nuevas ediciones.

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

---

# Contenido

<b>Figuras</b> . . . . .	v
<b>Tablas</b> . . . . .	vii
<b>Acerca de este manual</b> . . . . .	ix
A quién va dirigido este manual . . . . .	ix
Conocimientos previos necesarios. . . . .	ix
Términos utilizados en este manual . . . . .	ix
<b>Capítulo 1. Visión general</b> . . . . .	1
Notas para la Versión 1.0. . . . .	1
<b>Capítulo 2. Requisitos previos</b> . . . . .	3
<b>Capítulo 3. La familia MQSeries</b> . . . . .	5
<b>Capítulo 4. Requisitos</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 . . . . .	20
Colas . . . . .	20
Gestores de colas. . . . .	24
Operaciones de los gestores de colas . . . . .	26
Administración . . . . .	28
Mensajes de administración . . . . .	28
Administración selectiva. . . . .	30
Supervisión y otras acciones relacionadas. . . . .	31
Canales dinámicos . . . . .	31
Adaptadores . . . . .	32
Gestión de conexiones de marcación . . . . .	33
Rastreo . . . . .	33
Anotaciones de sucesos . . . . .	33
Redes MQSeries Everyplace . . . . .	33
Configuraciones y capacidad de crecimiento . . . . .	34
Entrega de mensajes asíncrona . . . . .	35
Entrega de mensajes síncrona . . . . .	36
Seguridad . . . . .	37
Seguridad local en MQSeries Everyplace . . . . .	38
Seguridad MQSeries Everyplace basada en colas . . . . .	38
Seguridad a nivel de mensajes . . . . .	40
El registro . . . . .	41
Entidades autenticables de MQSeries Everyplace . . . . .	41

El registro privado y las credenciales . . . . .	42
Registro automático . . . . .	42
Registro público y reproducción de certificados . . . . .	42
Utilización de los servicios de registro por parte de las aplicaciones . . . . .	43
Servicio de emisión de minicertificados por omisión . . . . .	43
La interfaz de seguridad. . . . .	44
Configuración y personalización . . . . .	44
Normas . . . . .	44
Tipos de conexiones . . . . .	46
Conexión de igual a igual . . . . .	47
Conexión cliente/servidor . . . . .	47
Tipos de conexiones diferentes . . . . .	47
Clases . . . . .	47
Carga de las aplicaciones . . . . .	48
<b>Capítulo 6. Redes MQSeries Everyplace y MQSeries . . . . .</b>	<b>49</b>
Interfaz con MQSeries . . . . .	49
Conversión de mensajes . . . . .	56
Función . . . . .	57
Compatibilidad . . . . .	57
Entrega segura. . . . .	58
<b>Capítulo 7. Aplicaciones y programas de utilidad . . . . .</b>	<b>59</b>
Postales . . . . .	59
MQSeries Everyplace Explorer . . . . .	59
<b>Capítulo 8. Interfaces de programación . . . . .</b>	<b>61</b>
<b>Apéndice. Avisos . . . . .</b>	<b>63</b>
Marcas registradas . . . . .	64
<b>Glosario . . . . .</b>	<b>65</b>
<b>Bibliografía. . . . .</b>	<b>69</b>
<b>Índice. . . . .</b>	<b>71</b>

---

## Figuras

1.	La familia MQSeries . . . . .	5
2.	Configuraciones sencillas de sistema principal y distribuida. . . . .	6
3.	Configuraciones típicas de estación de trabajo . . . . .	7
4.	Configuraciones típicas de dispositivos . . . . .	9
5.	Redes MQSeries Everyplace sencillas . . . . .	34
6.	Una red MQSeries Everyplace en estrella . . . . .	35
7.	Jerarquía de los objetos del puente MQSeries . . . . .	50





---

## Tablas

1.	Entornos de software soportados en la Versión 1 . . . . .	3
2.	Elementos de MQSeries Everyplace y MQSeries . . . . .	9
3.	Objetos de campos y propiedades de los campos de los que están compuestos . . . . .	17
4.	Propiedades de los objetos de atributo . . . . .	18
5.	Propiedades de los objetos de mensaje. . . . .	18
6.	Campos de objetos de mensaje a los que se da soporte . . . . .	19
7.	Propiedades de las colas . . . . .	23
8.	Propiedades de los gestores de colas locales. . . . .	26
9.	Propiedades de conexión (gestor de colas remoto) . . . . .	26
10.	Operaciones de mensajería en colas MQSeries Everyplace . . . . .	27
11.	Clases de mensajes de administración . . . . .	28
12.	Estructura genérica de un mensaje de administración . . . . .	29
13.	Soporte de autenticación, cifrado y compresión . . . . .	37
14.	Tipos de conexiones . . . . .	46
15.	Opciones de clases . . . . .	48
16.	Propiedades del objeto de puente . . . . .	51
17.	Propiedades del puente . . . . .	51
18.	Propiedades del proxy del gestor de colas MQSeries . . . . .	51
19.	Propiedades del servicio de conexión del cliente. . . . .	52
20.	Propiedades del escucha . . . . .	54
21.	Propiedades de las colas remotas de MQSeries . . . . .	55



---

## Acerca de este manual

Este manual es una introducción general a 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 Programming Reference* y la publicación *Guía de programación de MQSeries Everyplace*.

---

## A quién va dirigido este manual

Este manual va dirigido a todo aquel que esté interesado en utilizar un sistema de mensajería segura en dispositivos sencillos como, por ejemplo, sensores, teléfonos, PDA (Personal Digital Assistants) y portátiles.

---

## 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 segura.

Si no tiene estos conocimientos iniciales, es posible que la lectura de los siguientes manuales de MQSeries le resulte útil:

- *MQSeries An Introduction to Messaging and Queuing*
- *MQSeries para Windows NT® Versión 5.1 Comienzo rápido*

Estos manuales están disponibles 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 un grupo de productos MQSeries que se describen en el "Capítulo 3. La familia MQSeries" en la página 5.

### **Mensajería MQSeries**

hace referencia a los cuatro grupos de productos de mensajería que se describen en el "Capítulo 3. La familia MQSeries" en la página 5.

### **MQSeries**

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

- Mensajería distribuido
- Mensajería de sistema principal
- Mensajería de estación de trabajo

### **MQSeries Everyplace**

hace referencia al cuarto grupo de productos de mensajería de MQSeries.

### **Dispositivo**

Un sistema de cualquier tamaño que ejecuta programas MQSeries Everyplace pero que *no tiene instalado* ni un objeto de **gestor de canales MQSeries Everyplace** ni de **escucha de canal MQSeries Everyplace**.

**Nota:** Por motivos de licencia *dispositivo* es sinónimo de *cliente MQSeries Everyplace*.

### **Pasarela**

Un sistema de cualquier tamaño que ejecuta programas MQSeries Everyplace y que *tiene instalado* un objeto de **gestor de canales MQSeries Everyplace** o de **escucha de canal MQSeries Everyplace**.

**Nota:** Por motivos de licencia *pasarela* es sinónimo de *servidor MQSeries Everyplace*.

---

## Capítulo 1. Visión general

MQSeries Everyplace pertenece a la familia MQSeries de productos comerciales de mensajería de alta calidad. Se ha diseñado para poder satisfacer no sólo las necesidades de mensajería de dispositivos sencillos como por ejemplo, los sensores, los teléfonos, los PDA (Personal Digital Assistant) y los ordenadores portátiles, sino también para permitir conexiones móviles durante los desplazamientos y cubrir los requisitos que surgen cuando se utilizan redes de comunicaciones frágiles. Proporciona la función de entrega asegurada de mensajes de una sola vez que es estándar en MQSeries y permite intercambiar 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 sencillos necesitan que el subsistema de mensajería utilice de forma frugal los recursos del sistema y debido a esto se ha optimizado MQSeries Everyplace para mejorar la eficacia del protocolo y el espacio del sistema. Sus posibilidades no son idénticas a las de los demás miembros de la familia MQSeries que incorporan funciones de mensajería, pero permite interoperar sin problemas. MQSeries Everyplace incluye una gran gama de funciones de acceso a mensajes de forma remota, local, itinerante o móvil y también incluye funciones de seguridad y da soporte a la mensajería a través de redes poco seguras.

MQSeries Everyplace es miembro de la familia de sistemas generales IBM® y, por lo tanto, su diseño se integra perfectamente con los demás productos IBM inalámbricos y generales.

---

### Notas para la Versión 1.0

- La Versión 1.0 de MQSeries Everyplace es un kit de herramientas que permite a los usuarios escribir aplicaciones MQSeries Everyplace y crear un entorno en el que se pueden ejecutar.
- En este release, la utilización de MQSeries Everyplace en dispositivos generales es responsabilidad del proveedor de las aplicaciones y soluciones.



---

## Capítulo 2. Requisitos previos

La Tabla 1 muestra los entornos de software que se pueden utilizar para ejecutar MQSeries Everyplace Versión 1.<sup>1</sup>

*Tabla 1. Entornos de software soportados en la Versión 1*

	<b>Sistema operativo</b>
Dispositivo	EPOC
	Palm OS
	Windows CE
	Windows <sup>®</sup> 95 Windows 98 Windows NT v4 Windows 2000
Pasarela	Windows NT v4 Windows 2000
<b>Notas:</b>	
<ol style="list-style-type: none"><li>1. La Versión 1.0 se proporciona en Java para que se pueda utilizar en todas las plataformas que den soporte a Java.</li><li>2. Se dispone de un cliente con funciones limitadas, que sólo proporciona acceso síncrono a las colas remotas, como código base en C que únicamente se puede utilizar con los sistemas operativos conocidos como Palm OS.</li><li>3. Se recomienda Java 1.1, en el nivel más reciente que haya disponible. Consulte el sitio web de MQSeries Everyplace (<a href="http://www.ibm.com/software/mqseries/everyp1ace">www.ibm.com/software/mqseries/everyp1ace</a>) para obtener detalles sobre los niveles de Java probados.</li></ol>	

---

1. El código del dispositivo MQSeries Everyplace se puede ejecutar en cualquier dispositivo que ejecute Java<sup>®</sup>, pero únicamente se ha probado con los sistemas operativos que figuran listados en la Tabla 1.





## 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

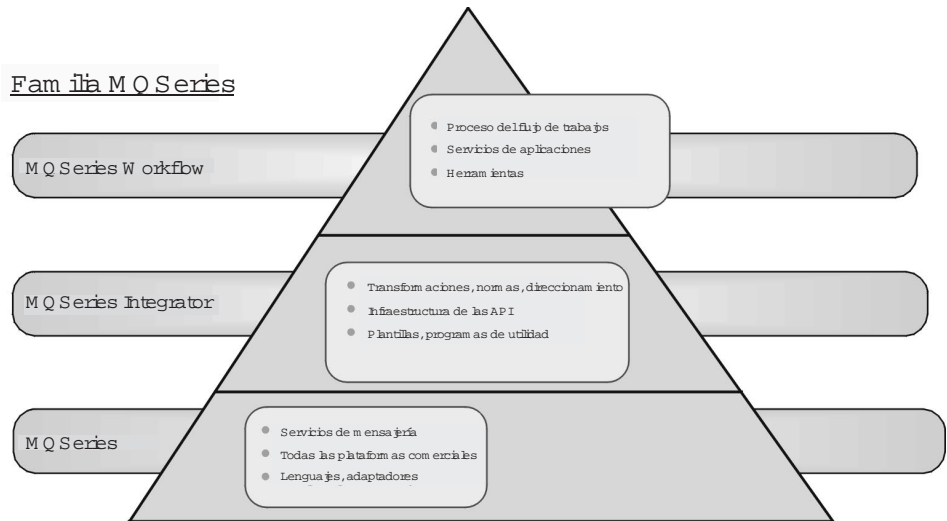


Figura 1. La familia MQSeries

- **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 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

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

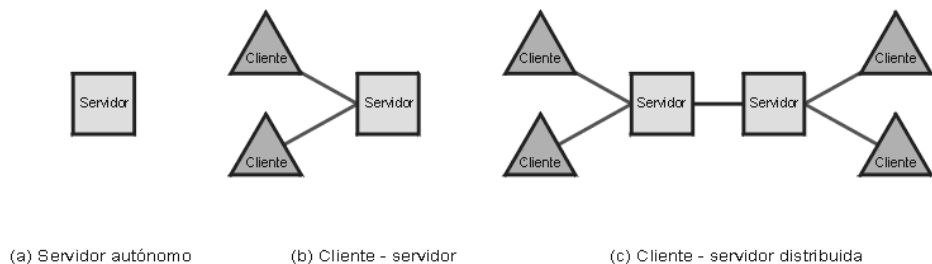
Las funciones de mensajería de la familia MQSeries las proporcionan los productos MQSeries (MQS) y MQSeries Everyplace, ya que cada uno de ellos está diseñado para dar soporte a una o más plataformas de servidor de hardware y a los sistemas operativos asociados. Dada la extensa variedad de soporte de plataformas, estos productos individuales se han organizado en grupos de productos, según su función y diseño comunes. Existen cuatro 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 de estación de trabajo:** MQSeries para Windows 3.1, 95, 98
- **Mensajería general:** MQSeries Everyplace

La mensajería por sí sola, independientemente del producto o grupo de productos determinado, está basado 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 local y obtienen o transfieren los mensajes a las colas. Si se transfiere un mensaje a un cola remota, esto es, una cola de la que es propietario un gestor de colas remoto, el mensaje se transmite a través de los canales al gestor de colas remoto. De este modo, los mensajes pueden pasar por uno o más gestores de colas intermediarios antes de llegar a su destino. El fundamento de la función de mensajería es 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.

Los productos de mensajería distribuida y de sistema principal de MQSeries se utilizan para dar soporte a muchas configuraciones de red y en todas ellas participan clientes y servidores <sup>2</sup>como puede observar en los ejemplos de las ilustraciones de la Figura 2.



*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 a 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

2. Tenga en cuenta que estos términos tienen significados muy específicos dentro de MQSeries que no siempre se corresponden con el uso que se les da comúnmente. En esta publicación se utilizan siempre dentro de la semántica específica de MQSeries.

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.

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.

Los productos de mensajería de estación de trabajo de MQSeries ofrecen un subconjunto de estas opciones de configuración. En lugar de dar soporte a servidores, estos productos dan soporte a estaciones de trabajo que tienen un gestor de colas aunque no dan soporte a conexiones de clientes. Sin embargo, las estaciones de trabajo se pueden conectar a otras estaciones de trabajo y también a servidores, a través de los canales de mensajes de MQSeries. De este modo, las estaciones de trabajo se consideran normalmente como servidores sencillos y sustituyen a los clientes cuando se necesitan posibilidades de intercambio asíncrono.

En la Figura 3 se muestran dos configuraciones de estación de trabajo típicas. En el ejemplo (b), las aplicaciones de estación de trabajo pueden ejecutarse independientemente de los servidores y de los clientes:

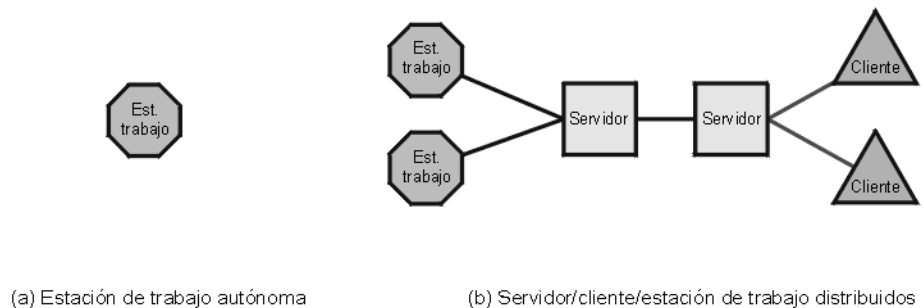


Figura 3. Configuraciones típicas de estación de trabajo

El producto MQSeries Everyplace con funciones de mensajería generales da soporte a configuraciones mediante el suministro de dispositivos y pasarelas.

El dispositivo MQSeries Everyplace es un sistema que ejecuta el código MQSeries Everyplace *sin un gestor de canales*. Esto significa que un dispositivo sólo puede comunicarse con otro dispositivo o pasarela cada vez. Los dispositivos MQSeries Everyplace pueden ser muy pequeños (como, por ejemplo, un sensor de un oleoducto), o dispositivos más grandes (como, por ejemplo, un teléfono, un PDA (Personal Data Assistant) o un ordenador portátil) e incluso sistemas de sobremesa y estaciones de

trabajo. Normalmente, este tipo de dispositivos se denominan dispositivos generales, aunque el término implica una limitación en cuanto al tamaño y a las posibilidades que, en realidad, no existen en el producto.

Una *pasarela* es un sistema que ejecuta el código MQSeries Everyplace *que tiene configurado un gestor de canales MQSeries Everyplace o un escucha de canal MQSeries Everyplace*. Esto ofrece todas las posibilidades del código de dispositivo y además permite entablar comunicaciones con diferentes pasarelas de dispositivo al mismo tiempo. Las pasarelas también proporcionan un mecanismo que permite intercambiar mensajes entre una red MQSeries Everyplace y una red MQSeries.

En primer término, los dispositivos combinan muchos de los atributos de clientes y servidores. Se pueden configurar de modo que tengan todas las posibilidades de transferencia a colas y puedan operar de forma asíncrona. También pueden acceder a las colas remotas, una característica que tiene alguna similitud con el acceso del cliente a las colas del servidor. A diferencia de los servidores, los dispositivos no se pueden conectar a clientes. Los dispositivos pueden establecer comunicaciones entre sí directamente, mediante la posibilidad de mensajería de igual a igual. Los dispositivos también pueden comunicarse a través de canales aunque estos canales son exclusivos de MQSeries Everyplace y se denominan *canales dinámicos* para diferenciarlos de los canales de cliente de MQSeries y de los canales de mensajes de MQSeries. Los canales dinámicos son bidireccionales y dan soporte a toda la gama de funciones que proporciona MQSeries Everyplace, incluidas la mensajería síncrona y asíncrona.

Necesariamente, las pasarelas dan soporte a los canales dinámicos de MQSeries Everyplace para poder comunicarse con los dispositivos. Opcionalmente, pueden dar soporte a los canales de cliente de MQSeries para poder comunicarse con los servidores. Al igual que éstos, las pasarelas tienen gestores de colas y, por lo tanto, dan soporte a las aplicaciones de mensajería locales.

En la Figura 4 en la página 9 se muestran algunas configuraciones generales típicas.

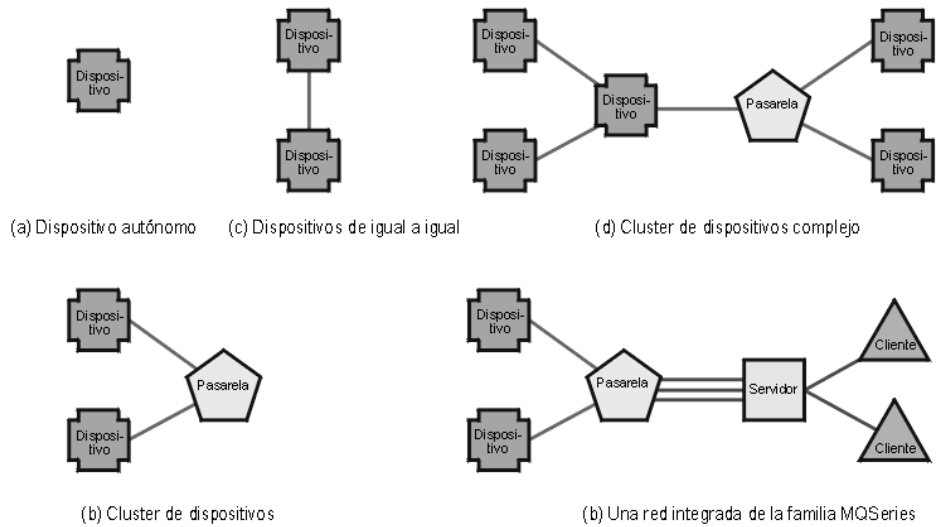


Figura 4. Configuraciones típicas de dispositivos

En la Figura 4 (b) se utiliza una pasarela para conectar los dispositivos entre sí. Una característica de las pasarelas es que pueden manejar simultáneamente varias peticiones de conexión de entrada recibidas al mismo tiempo, a diferencia de los dispositivos que únicamente pueden manejar una petición de este tipo cada vez. Tanto las pasarelas como los dispositivos pueden emitir simultáneamente varias peticiones de salida. Si en la configuración (b) se utilizara un dispositivo en lugar de la pasarela, entonces dos dispositivos de terminal tendrían que turnarse para contactar con este dispositivo intermediario, a pesar de que éste podría contactar con los dos al mismo tiempo. En la configuración (d) se utilizan tanto un dispositivo como una pasarela para enlazar los dispositivos. En (e) se utiliza una pasarela para enlazar una red de dispositivos a un servidor MQSeries, una configuración en la que pueden fluir los mensajes entre todos los componentes, esto es, dispositivos, pasarelas, servidores, estaciones de trabajo y clientes. En la Tabla 2 se muestran las características más importantes de estos componentes.

Tabla 2. Elementos de MQSeries Everyplace y MQSeries

Componente	Características	Lo proporciona
Dispositivo (MQSeries Everyplace )	<p>Proporciona funciones de mensajería seguras a aplicaciones a través de canales dinámicos</p> <ul style="list-style-type: none"> <li>permite el acceso síncrono a colas remotas y locales</li> <li>permite la entrega de mensajes asíncrona a colas remotas</li> <li>sólo puede manejar las peticiones recibidas de una en una</li> </ul>	Sistemas generales

Tabla 2. Elementos de MQSeries Everyplace y MQSeries (continuación)

Componente	Características	Lo proporciona
Cliente (MQS)	<p>Proporciona funciones de mensajería seguras a aplicaciones locales</p> <ul style="list-style-type: none"> <li>requiere una conexión síncrona de un canal de cliente con un servidor</li> <li>permite el acceso síncrono únicamente a las colas del servidor con el que se ha conectado</li> <li>permite la entrega de mensajes asíncrona a las colas remotas a través del servidor con el que se ha conectado</li> </ul>	Sistema principal distribuido
Pasarela (MQSeries Everyplace )	<p>Proporciona funciones de mensajería seguras a aplicaciones a través de canales dinámicos</p> <ul style="list-style-type: none"> <li>permite el acceso síncrono a colas remotas y locales</li> <li>permite la entrega de mensajes asíncrona a colas remotas</li> <li>puede manejar varias peticiones recibidas al mismo tiempo</li> </ul> <p>Puede soportar la conexión de varios servidores MQSeries a través de los canales de clientes</p>	Sistemas generales
Servidor (MQS)	<p>Proporciona funciones de mensajería seguras a aplicaciones a través de canales de mensajes</p> <ul style="list-style-type: none"> <li>permite el acceso síncrono a colas locales</li> <li>permite la entrega de mensajes asíncrona a colas remotas</li> </ul> <p>Puede soportar la conexión de varios clientes MQSeries a través de los canales de clientes</p>	Sistema principal distribuido
Estación de trabajo (MQS)	<p>Proporciona funciones de mensajería seguras a aplicaciones a través de canales de mensajes</p> <ul style="list-style-type: none"> <li>permite el acceso síncrono a colas locales</li> <li>permite la entrega de mensajes asíncrona a colas remotas</li> </ul>	Estación de trabajo

---

## Capítulo 4. Requisitos

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 sencillos como por ejemplo, PDA, teléfonos y sensores, y de este modo les permite participar en una red de mensajería de MQSeries. También da soporte a dispositivos intermedios como, por ejemplo, ordenadores portátiles, estaciones de trabajo y determinadas plataformas distribuidas. MQSeries Everyplace ofrece la misma calidad de servicio, en la entrega segura de mensajes de una sola vez, y permite el intercambio de mensajes con otros miembros de la familia.
- 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. 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. 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

No existe una lista que limite las aplicaciones posibles de MQSeries Everyplace ya que las opciones son muchas y variadas, pero se supone que un gran número de éstas serán aplicaciones personalizadas desarrolladas para grupos de usuarios determinados. La lista siguiente ofrece algunos de los ejemplos que se han tenido en cuenta:

- **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 influenciado 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:** considerablemente por debajo de 100.000 bytes para un sistema de dispositivo con una configuración mínima.
- **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.



---

## 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. Tanto los dispositivos como las pasarelas almacenan los datos de configuración en un registro.

Las aplicaciones de los dispositivos pueden utilizar cualquiera o todas las API o funciones que estén disponibles en el dispositivo y no se han de limitar a utilizar las interfaces de programación de MQSeries Everyplace. Los dispositivos MQSeries Everyplace se pueden conectar, a través de los canales dinámicos, a otros dispositivos MQSeries Everyplace y a una pasarela MQSeries Everyplace.

Las aplicaciones de las pasarelas también pueden utilizar cualquiera o todas las API o funciones que haya disponibles en la pasarela y no se han de limitar a utilizar las interfaces de programación de MQSeries. Una pasarela se puede conectar, a través de los canales dinámicos, a otras pasarelas y a dispositivos MQSeries Everyplace. Mediante los canales de cliente MQSeries, una pasarela se puede conectar a uno o varios servidores MQSeries (pero no se puede conectar a otras pasarelas MQSeries Everyplace). MQSeries Everyplace y MQSeries pueden coexistir en una sola máquina, aunque la presencia de MQSeries es totalmente opcional.

Las posibilidades de todas las funciones de los dispositivos y de las pasarelas son las mismas, con las siguientes excepciones:

- Las pasarelas pueden manejar varias peticiones de entrada recibidas al mismo tiempo (de otros dispositivos o pasarelas)
- Las pasarelas pueden intercambiar datos al mismo tiempo con varios servidores MQSeries

Los canales dinámicos dan soporte a las siguientes conexiones de red:

- Conexiones de marcación
- Conexiones permanentes, por ejemplo una LAN convencional, una línea de abonado, una LAN sin cables o de infrarrojos.

Los protocolos de comunicaciones se implementan mediante un conjunto de adaptadores, uno para cada protocolo soportado. Esto permite añadir nuevos protocolos cuando es necesario y adaptar el espacio de memoria de un entorno dado a una configuración determinada.

Los colas se correlacionan individualmente con el soporte de almacenamiento mediante otro conjunto de adaptadores. De este modo, una cola se puede almacenar en el sistema de archivos o en la memoria dependiendo del adaptador que se haya seleccionado.

La interfaz de programación de MQSeries Everyplace se ha diseñado de modo que puedan escribirse aplicaciones independientemente de la ubicación de las colas. Así, un programa que se haya diseñado para acceder a las colas locales podrá ejecutarse sin modificaciones desde un gestor de colas remoto (siempre que cumpla los requisitos de seguridad que estén en vigor y que acepte que no se da soporte a determinadas operaciones MQSeries Everyplace en las colas remotas). Esta independencia se aplica a todas las funciones administrativas.

---

## Objetos de mensaje

Los objetos de mensaje de MQSeries Everyplace presentan unas diferencias fundamentales en relación con los mensajes que soporta MQSeries. En MQSeries, un mensaje es una matriz de bytes, dividida en una cabecera de mensaje y un texto de mensaje. MQSeries comprende la cabecera de mensaje que contiene información importante como la identidad de la cola de respuestas, el gestor de colas de respuestas, el id de mensajes y el id de correlación, pero el texto del mensaje no lo 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*. Los mensajes son objetos verdaderos, sin ningún concepto de cabecera ni de texto de mensaje. La naturaleza verdadera del objeto de mensaje resulta más fácil de asimilar cuando se comprende qué es el objeto de campos base. Estos objetos de campos, se utilizan comúnmente en MQSeries Everyplace y 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 tipos de campos pueden ser:

- **ASCII** una serie de caracteres o una matriz dinámica de series de caracteres ASCII
- **Booleano** un valor booleano
- **De bytes** una matriz fija o una matriz dinámica de valores de byte
- **De coma flotante doble** una matriz fija o una matriz dinámica de valores de coma flotante dobles
- **De objetos de campos** objetos de campos o una matriz dinámica de objetos de campos (con lo cual se da soporte a objetos de campos anidados)
- **De coma flotante** una matriz fija o una matriz dinámica de valores de coma flotante
- **De enteros** (de 4 bytes), una matriz fija o una matriz dinámica de enteros
- **De enteros largos** (de 8 bytes), una matriz fija o una matriz dinámica de enteros largos
- **De enteros cortos** (de 2 bytes), una matriz fija o una matriz dinámica de enteros cortos

- **UNICODE** una serie de caracteres o una matriz dinámica de series de caracteres UNICODE

Los objetos de campos tienen un *tipo* y éste se corresponde con el nombre de la clase de objeto de programación. Los programas de aplicación utilizan los descendientes de esta clase de objeto como objetos de mensaje y el tipo lo utiliza MQSeries Everyplace para convertir en instancia la clase de objeto cuando es necesario, por ejemplo, después de que un objeto de mensaje ha fluido a través de un canal.

Los objetos de campos proporcionan diferentes métodos, por ejemplo, se pueden enumerar los campos o se puede verificar su existencia. Del mismo modo, se puede comparar la igualdad de los objetos de campo. Presentan la posibilidad de volcar y restaurar los elementos de una matriz de bytes, que se utiliza para, por ejemplo, proporcionar los datos para transmitir el objeto a través de un enlace y para restaurar el objeto después de su transmisión. Los métodos de vuelco y restauración se pueden alterar temporalmente para permitir que los objetos de campos se conviertan automáticamente en series de caracteres de otro tipo y así poder, por ejemplo, consultar en una base de datos el contenido de sus campos durante la transmisión. En la Tabla 3 se muestran las propiedades de los objetos de campos y los campos de los que están compuestos.

*Tabla 3. Objetos de campos y propiedades de los campos de los que están compuestos*

Propiedad	Disponibilidad	
	Objetos de campos	Campos
Objeto de atributo asociado	Opcional	
Campo(s) que los componen	sí	
Oculto		sí
Nombre		sí
Tipo	sí	sí
Valor		sí

Un objeto que tiene una propiedad de campo oculto permite ignorar este campo cuando se realiza una operación de comparación.

Los *objetos de atributo* contienen los mecanismos que permiten realizar las funciones de autenticación, cifrado y compresión y pueden estar asociados a los objetos de campos.

- **Autenticación:** controla el acceso
- **Compresión:** disminuye los requisitos de memoria (tanto en la transmisión como en el almacenamiento)
- **Cifrado:** protege el contenido cuando se realiza un vuelco del objeto (y permite su restauración)

Los objetos de atributo son la base del modelo de seguridad de MQSeries Everyplace, permiten un acceso selectivo al contenido y protegen los datos en el soporte de

almacenamiento de copias de seguridad, entre otros usos. La Tabla 4 lista las propiedades de los objetos de atributo. Cuando el valor de la norma está disponible, controla qué operaciones están permitidas.

*Tabla 4. Propiedades de los objetos de atributo*

Propiedad	Disponibilidad
Cifrador	opcional (es posible que sea necesario en algunas circunstancias)
Autenticador	opcional
Compresor	opcional
Norma	opcional
Tipo	opcional

Los objetos de mensaje se derivan de los objetos de campos e incluyen un identificador exclusivo (UID) que genera MQSeries Everyplace. Este UID identifica de forma exclusiva a un objeto de mensaje y consta 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)

Los objetos de mensaje poseen las propiedades básicas que se listan en la Tabla 5 más las que heredan como objetos de campos.

*Tabla 5. Propiedades de los objetos de mensaje*

Propiedad	Descripción
Msg_OriginQMgr	El nombre del gestor de colas que ha enviado el mensaje
Msg_Time	La hora en que la aplicación ha creado el objeto de mensaje

Estas dos propiedades forman el identificador exclusivo (UID) del objeto de mensaje.

No es necesario ningún otro tipo de información en un mensaje que vaya dirigido a otro gestor de colas MQSeries Everyplace, aunque normalmente se incluyen otros campos para transportar el contenido informativo de los mensajes. Generalmente los mensajes son sucesores de la clase de objeto del mensaje basey, por lo tanto, contienen campos adicionales según su finalidad. Entre estos campos adicionales, hay varios que son comunes a una amplia gama de aplicaciones como, por ejemplo, "gestor de colas de respuestas". Y para ello MQSeries Everyplace proporciona medidas que permiten dar soporte a los mismos.

La Tabla 6 lista los campos soportados.

*Tabla 6. Campos de objetos de mensaje a los que se da soporte*

Nombre de campo	Utilización
Msg_CorreIID	Serie de bytes que se utiliza normalmente para correlacionar una respuesta con el mensaje original
Msg_ExpireTime	Hora después de la cual se puede suprimir el mensaje (incluso si no se ha entregado)
Msg_LockID	La clave necesaria para desbloquear un mensaje
Msg_MsgID	Lo utiliza la aplicación para la correlación con el mensaje original
Msg_Priority	Prioridad del mensaje
Msg_ReplyToQ	El nombre de la cola a la que debe dirigirse una respuesta al mensaje
Msg_ReplyToQMgr	El nombre del gestor de colas al que debe dirigirse una respuesta al mensaje
Msg_Resend	Indica que el mensaje es un reenvío de un mensaje anterior
Msg_Style	Distingue los mandatos de petición/respuesta, etc.

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 el 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, la hora de caducidad activa la caducidad del mensaje, etc.

Los objetos de mensaje tienen definidos varios métodos, por ejemplo, la posibilidad de extraer el UID de mensaje, el nombre del gestor de colas de origen y la hora de creación del objeto. Otros métodos útiles los heredan de la clase de objeto de campos, por ejemplo, los diferentes métodos para obtener y transferir los valores de los campos. El método que posee un interés especial es el método de *vuelco*, que se utiliza para realizar un vuelco de los datos del objeto en una serie de bytes. MQSeries Everyplace llama a este método cuando se ha de guardar un mensaje en un soporte de almacenamiento permanente y cuando se ha de transmitir a través de un canal dinámico. Con este método el objeto de mensaje por sí solo es el responsable de determinar la representación externa de su valor de datos y esto puede utilizarse de varias formas. Por ejemplo, un objeto puede volcar simplemente los valores de los campos que lo forman o puede optar por consultarlos en una base de datos. El método de *restauración* complementario ofrece posibilidades de control similares cuando se vuelve a crear un objeto a partir de su formato volcado. Tenga en cuenta que si el objeto de mensaje tiene asociado un objeto de atributo, durante el vuelco se invocan las propiedades de codificación, cifrado y compresión de los datos de atributo, del mismo modo que durante el proceso de restauración se invocan las propiedades de decodificación, descifrado y descompresión.

Cuando MQSeries Everyplace envía un objeto de mensaje a través de un flujo, para no colapsar la red con cable, no transporta el archivo de clase asociado. Al mismo tiempo,

la clase de mensaje correspondiente debe estar disponible en cada uno de los gestores de colas en los que se creará una instancia del objeto de mensaje.

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

## 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 de 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 dentro de un rango de 0 a 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 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. Las aplicaciones no pueden acceder directamente al objeto de cola,<sup>3</sup> sino que 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>4</sup> pero debe ser exclusivo en un gestor de colas determinado.

---

3. El acceso directo sólo se permite en base a una norma de cola.

4. 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.



MQSeries Everyplace da soporte a diferentes tipos de colas:

### **Colas locales**

Las colas locales las utilizan las aplicaciones para almacenar mensajes de forma segura y fiable. Poseen un almacenamiento de mensajes al que se accede a través de una clase de adaptador, generalmente la clase de adaptador de disco. Sin embargo, con MQSeries Everyplace se suministra una clase de adaptador de memoria que alberga en la memoria el almacenamiento de mensajes con lo que se obtiene un acceso rápido (aunque supone la pérdida de los mensajes si se produce una anomalía general del sistema). Al crear el adaptador adecuado, los mensajes se pueden almacenar en cualquier lugar, tanto cola por cola, como podría ser el caso de una base de datos relacional, como grabándolos en un CD, etc. Las colas locales se pueden utilizar tanto en línea como fuera de línea, esto es, tanto si están conectadas como si no están conectadas a una red. El acceso y la seguridad son propiedad de la cola y su uso puede otorgarlo un gestor de colas remoto, cuando está conectado a una red, con lo que permite a otros enviar o recibir mensajes a y desde la cola. El acceso a la cola local siempre es síncrono.

### **Colas remotas**

Las colas remotas no residen en el entorno local, sino que localmente existe una definición que identifica al gestor de colas que la posee y a la cola real. Se puede acceder a las colas remotas de forma síncrona o asíncrona. Si localmente existe una definición de la cola remota, entonces la modalidad de acceso estará basada en esta definición. De no ser así, se produce una detección de la cola, de tal modo que se detectan las características y la modalidad de acceso obligatoriamente ha de ser síncrona.

Las colas *síncronas* son colas a las que únicamente se puede acceder cuando se está conectado a una red que tiene una vía de acceso al gestor de colas propietario. Si la red no está establecida, entonces las operaciones de transferencia, obtención y examen, (consulte la Tabla 10 en la página 27) generarán 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 recepción de mensajes, y en tal caso, MQSeries Everyplace deja de ser responsable de la entrega de mensajes segura de una sola vez.

Las colas *asíncronas* son colas a las que se puede transferir mensajes pero de las que no se pueden recuperar. Si se establece una conexión de red, entonces los mensajes se envían al gestor de colas propietario y a la cola. Sin embargo, si la red no está conectada, los mensajes se almacenarán localmente hasta que se establezca una conexión de red y pasen a transmitirse. De este modo, las aplicaciones pueden realizar operaciones cuando el dispositivo está fuera de línea. Sin embargo, es necesario que estos tipos de colas posean un almacenamiento de mensajes donde guardar temporalmente los mensajes.

### **Colas para almacenar y remitir**

Este tipo de cola almacena mensajes hasta que se pueden remitir al siguiente gestor de colas (no necesariamente el propietario). Normalmente, aunque no necesariamente, este tipo de colas están definidas en una pasarela y el

dispositivo tendrá que recoger sus mensajes cuando se conecte a la red. Las colas para almacenar y remitir pueden contener mensajes para muchos gestores de colas de destino o puede haber una cola para almacenar y remitir por gestor de colas de destino. Cuando un emisor desea enviar un mensaje a un receptor que puede estar desconectado, a pesar de todo, el emisor puede enviar el mensaje al gestor de colas/cola del receptor y el servidor intermediario detectará que el receptor no está conectado y almacenará el mensaje en su almacenamiento local de mensajes. La aplicación emisora no necesita realizar ninguna modificación para enviar un mensaje a una cola de destino que esté conectada o desconectada.

### **Colas del servidor inicial**

Normalmente este tipo de colas reside en un dispositivo (que se supone que está conectado ocasionalmente) y apunta a una cola para almacenar y remitir de un gestor de colas al que se conoce como el servidor inicial. La cola del servidor inicial extrae los mensajes del servidor inicial en cuanto el dispositivo se conecta a la red. Cuando la cola extrae un mensaje del servidor lo transfiere al gestor de colas local utilizando las llamadas de los métodos `putMessage` y `confirmputMessage` (consulte el apartado “Operaciones de los gestores de colas” en la página 26). A continuación, es responsabilidad del gestor de colas transferir el mensaje a la cola local correcta. Este método de extracción, que permite obtener los mensajes desde el 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 y una segunda petición/respuesta para el flujo de confirmación. Las colas del servidor inicial normalmente tienen 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.

### **Colas de administración**

Este tipo de colas recibe los mensajes de administración de MQSeries Everyplace. También se puede utilizar una cola de respuestas a mensajes de administración opcional para recibir las respuestas a los mensajes de administración que envíe el sistema MQSeries Everyplace. Las colas de administración no comprenden cómo se han de realizar las tareas de administración, simplemente manejan los mensajes donde están encapsulados los detalles de administración.

### **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 proporciona un transformador muy 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 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. Alternativamente, 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 7 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 7. 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	El nombre del puente MQ propietario
Queue_ClientConnection	El nombre de la conexión del cliente
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
Queue_Expiry	La hora de caducidad de los mensajes
Queue_FileDesc	La ubicación en la que se almacena la cola
Queue_MaxMsgSize	La longitud máxima permitida para los mensajes de la cola
Queue_MaxQSize	El número máximo de mensajes permitido

Tabla 7. Propiedades de las colas (continuación)

Propiedad	Descripción
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 se 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_RemoteQName	El nombre del campo MQSeries remoto
Queue_Rule	La clase de norma para las operaciones de cola
Queue_TargetRegistry	El tipo de registro de la cola
Queue_Transporter	La clase del transportador
Queue_Transformer	La clase del transformador

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

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 tipo correcto 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 segura. 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 puede enviar este tipo de mensajes y que durante su transporte el contenido está fuertemente protegido. No se permitirá que ninguna aplicación acceda a la cola y ninguna será necesaria.

---

## Gestores de colas

El gestor de colas de MQSeries Everyplace proporciona a las aplicaciones acceso a los mensajes y las colas y controla los canales. En MQSeries Everyplace Versión 1.0 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> Los gestores de colas se pueden configurar con o sin colas locales. Todos los gestores de colas

---

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.

soportan las operaciones de mensajería síncronas. Un gestor de colas que permite la colocación en colas locales también da soporte a la entrega de mensajes asíncrona.

Las entregas de mensajes asíncronas y síncronas tienen diferentes características y consecuencias:

#### **Entrega de mensajes asíncrona <sup>7</sup>**

En la entrega de mensajes asíncrona, la aplicación pasa el mensaje a MQSeries Everyplace para su entrega a una cola remota. Inmediatamente después de la operación de transferencia regresa a la aplicación. MQSeries Everyplace conserva el mensaje localmente y de forma temporal hasta que puede realizarse la entrega. La entrega puede realizarse en fases, y MQSeries Everyplace es el responsable de la entrega. Esta modalidad de operación proporciona el tipo de *entrega segura de una sola vez*. Consulte el apartado “Entrega de mensajes asíncrona” en la página 35 donde se describe más detalladamente.

#### **Entrega de mensajes síncrona:**

La mensajería síncrona se puede utilizar en:

- Colas de destino de un gestor de colas MQSeries Everyplace direccionadas a través de una red MQSeries Everyplace
- Colas de destino de un gestor de colas MQSeries conectadas directamente a una pasarela MQSeries Everyplace
- Colas de destino de un servidor MQSeries conectado indirectamente

En la mensajería síncrona, la aplicación transfiere el mensaje a MQSeries Everyplace para su entrega. MQSeries Everyplace contacta de forma síncrona a la cola remota de destino y coloca en ésta el mensaje. Después de la entrega MQSeries Everyplace regresa a la aplicación.

El contacto con el gestor de colas remoto puede requerir el direccionamiento de MQSeries Everyplace a través de dispositivos intermedios y de pasarelas. Consulte el apartado “Entrega de mensajes síncrona” en la página 36 donde se describe más detalladamente.

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 local 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.

---

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.

7. MQSeries Everyplace no distingue entre las modalidades permanentes y no permanentes que ofrece MQSeries, sino que únicamente da soporte a la modalidad permanente. Sin embargo, si es necesario, hay una opción de almacenamiento intermedio en cola que permite elegir entre un mejor rendimiento o la recuperación.

Un gestor de colas local posee propiedades que reflejan la gestión local de las colas. También necesita una *definición de conexión* para cada gestor de colas remoto con el que debe ponerse en contacto. De aquí que, algunas veces, se haga referencia a las definiciones de conexión como *definiciones de gestores de colas remotos*. Estas definiciones pueden incluir toda la información necesaria para una comunicación directa entre los gestores de colas (incluida la dirección de red) o simplemente pueden indicar que la comunicación es indirecta, esto es, que se realiza a través de un gestor de colas intermediario. En este último caso, lo único que se necesita es el nombre del gestor de colas del salto siguiente.

En la Tabla 8 y en la Tabla 9 se muestran las propiedades de los gestores de colas.

*Tabla 8. Propiedades de los gestores de colas locales*

Propiedad	Descripción
QMgr_ChnlAttrRules	Las normas de los atributos de canal
QMgr_ChnlTimeout	El tiempo excedido del canal
Admin_Class	La clase del gestor de colas
QMgr_Description	La descripción Unicode
Admin_Name	El nombre del gestor de colas
QMgr_Rules	La clase de norma para las operaciones del gestor de colas

*Tabla 9. Propiedades de conexión (gestor de colas remoto)*

Propiedad	Descripción
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 de la servlet)
Con_Aliases	Los nombres alternativos del gestor 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
Admin_Name	El nombre del gestor de colas

En una definición de conexión se soportan varios adaptadores.

## Operaciones de los gestores de colas

Los gestores de colas soportan las operaciones de mensajería y, opcionalmente, la gestión de las colas. Las aplicaciones pueden acceder a los mensajes a través de las operaciones que se realizan en un gestor de colas. A menos que se especifique un filtro, se recupera el primer mensaje que hay disponible. Un filtro es un objeto de campo con el que se buscan coincidencias y para realizar una recuperación selectiva se pueden utilizar cualquiera de los campos del mensaje. La operación de obtención, al

igual que todas las operaciones de envío y recuperación, puede tener opcionalmente un objeto de atributo que se utilizará para codificar y decodificar un mensaje.

En MQSeries Everyplace, al igual que en MQSeries, una operación de obtención es normalmente destructiva. Si se necesita una mensajería segura entre MQSeries Everyplace y la aplicación, deberá utilizarse la secuencia del método en que se realiza una operación de obtención seguida por una confirmación. En primer lugar se emite una obtención con un id de confirmación (su valor lo selecciona la aplicación), esta operación obtiene el mensaje 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 forma 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 bloqueo 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*. Alternativamente 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 10

Tabla 10. Operaciones de mensajería en colas MQSeries Everyplace

	Cola local	Cola remota <sup>1</sup>	
		Síncrona	Asíncrona
Examinar (±bloqueo, ±filtro)	Sí	Sí	
Eliminar	Sí	Sí	
Obtención (±filtro)	Sí	Sí	
Escucha (±filtro)	Sí		
Transferencia	Sí	Sí	Sí
Espera (±filtro)	Sí	Sí	

Tabla 10. Operaciones de mensajería en colas MQSeries Everyplace (continuación)

	Cola local	Cola remota <sup>1</sup>	
		Síncrona	Asíncrona
<b>Notas:</b>			
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 entre MQSeries Everyplace y MQSeries que se suministra con MQSeries Everyplace Versión 1.0 sólo da soporte a las operaciones de transferencia.			

Opcionalmente, los gestores de colas pueden cargar aplicaciones (clases) inmediatamente después de su inicio, del mismo modo que pueden finalizar las aplicaciones al concluir. Los gestores de colas generan sucesos que reflejan el estado o una condición de error, estos sucesos aparecen por omisión en las anotaciones de sucesos.

## 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 MQSeries Everyplace Explorer<sup>8</sup>, una herramienta de gestión que proporciona una interfaz gráfica de usuario para las tareas de 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 11 en la página 29 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 MQQueueAdminMsg. El puente entre MQSeries Everyplace y MQSeries utiliza subclases de MQAdminMsg de este modo.

8. MQSeries Everyplace Explorer no está incluido en la Versión 1.0 pero se puede bajar desde el sitio de software de MQSeries en la World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>).



Tabla 11. Clases de mensajes de administración

Clases de mensajes de administración	Utilización
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
MQeStoreAndForwardQueueAdminMsg	Administración de las colas para almacenar y remitir
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 MQS
MQeListenerAdminMsg	Administración de un objeto de escucha de colas de transmisión, que se utiliza como puente para recoger mensajes de MQS
MQeBridgeAdminMsg	Administración de un puente con MQS
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 MQ

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 12:

Tabla 12. Estructura genérica de un mensaje de administración

Campos de nivel 1	Campos de nivel 2 e inferiores	Utilización
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

Tabla 12. Estructura genérica de un mensaje de administración (continuación)

Campos de nivel 1	Campos de nivel 2 e inferiores	Utilización
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
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.

Además del soporte de mensajes de administración básico, también se proporcionan clases de ayuda que encapsulan la construcción del mensaje y la interpretación de la respuesta. Opcionalmente, estas clases pueden suministrar diálogos de usuario, con lo que también pueden servir para crear herramientas de administración sencillas.<sup>9</sup>

## 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

9. Estas clases no se incluyen en la Versión 1.0 pero pueden bajarse desde el sitio de software de MQSeries en la World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>).

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 de seguridad). 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 44).

## Supervisión y otras acciones relacionadas

A menudo, las tareas de administración van más allá del ámbito de creación y modificación de objetos, por ejemplo, cuando se supervisa el estado del sistema y se manejan situaciones de errores, lo que implica la realización de acciones como informar a un operador de que una cola está prácticamente llena o ejecutar la acción correspondiente cuando llega un mensaje demasiado grande para su cola de destino. Estos aspectos los maneja MQSeries Everyplace 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 44).

---

## Canales dinámicos

MQSeries Everyplace se comunica con gestores de colas de pasarelas y dispositivos mediante 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. 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 cluster* de MQSeries tienen algunas características similares a las de canales dinámicos de MQSeries Everyplace, pero existen algunas diferencias.

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:

- **Autenticador:** puede ser nulo o un objeto *autenticador* 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 autenticador 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. Este tipo de cifrado no es seguro pero hace que los datos no puedan visualizarse. 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 autenticadores, compresores y objetos cifradores que puede consultar en la Tabla 13 en la página 38.

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.

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 segura, 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, de TCP/IP nativo y de 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. Lo que significa que no es necesario ningún soporte explícito para las redes de marcación en las implementaciones de dispositivos de MQSeries Everyplace.

---

## 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.

---

## Redes MQSeries Everyplace

Las redes MQSeries Everyplace están conectadas a los dispositivos y a las 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.

## Configuraciones y capacidad de crecimiento

En la Figura 5 se muestra una selección de las topologías de red básicas de MQSeries Everyplace. Para este fin se supone que cada una de ellas se ha configurado con posibilidades de comunicación síncrona y asíncrona.

El caso más sencillo es aquel en el que un dispositivo autónomo soporta la comunicación síncrona entre aplicaciones a través de las colas locales, como en el caso (a) de la parte superior. Sin embargo, el caso más interesante es el caso (b), en el que se ilustra una red de igual a igual. En este caso, es necesario que ambos dispositivos utilicen el mismo protocolo de comunicaciones y que como mínimo uno de los dispositivos esté configurado con posibilidades de escucha, para que pueda responder al otro que intenta realizar el contacto. Obviamente, en este caso sencillo, la comunicación únicamente es posible cuando ambos están disponibles en la misma red. La mensajería asíncrona permite que las aplicaciones se ejecuten cuando los dispositivos no están conectados, pero la mensajería síncrona sólo es posible cuando los dispositivos están conectados.

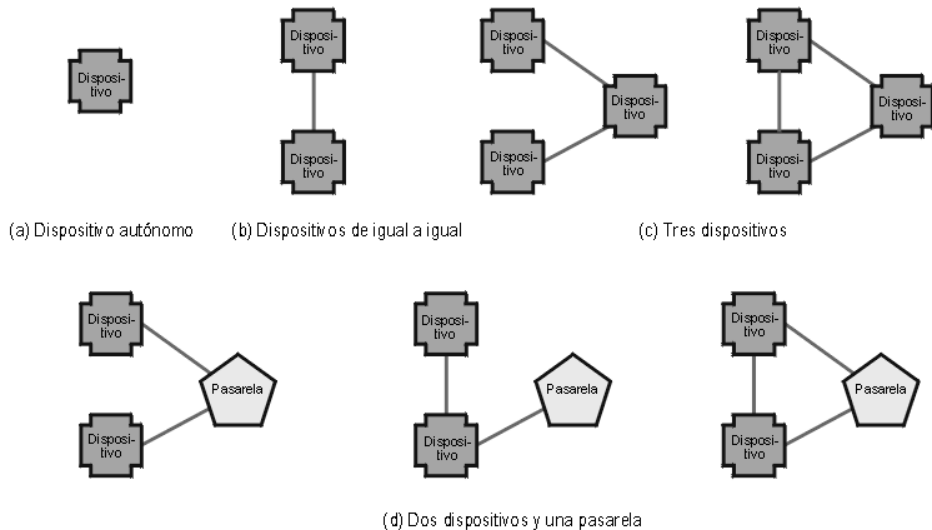


Figura 5. Redes MQSeries Everyplace sencillas

Con tres dispositivos existen muchas más posibilidades. Cualquiera de ellos puede realizar la función de dispositivo de enlace y los tres pueden estar conectados entre sí. Si se configuran para que puedan intercambiar mensajes entre sí de igual a igual, entonces se aplican las consideraciones mencionadas. Si se comunican a través de un tercer dispositivo, entonces la flexibilidad es mucho mayor. No es necesario que los dos dispositivos de comunicaciones utilicen el mismo protocolo de comunicaciones, cada uno de ellos deberá utilizar uno que pueda comprender el dispositivo que actúa como puente. La conectividad asíncrona permite además que los mensajes fluyan incluso si el emisor y el receptor nunca están en la red al mismo tiempo (siempre y cuando uno de ellos esté en la red en el momento en que el dispositivo de enlace esté disponible). La comunicación síncrona requiere que los tres dispositivos estén

disponibles al mismo tiempo. Para los dispositivos que suelen estar desconectados, una configuración que pase siempre a través de un tercer nodo de algún tipo resulta muy adecuada, siempre y cuando el intermediario esté normalmente disponible.

En la práctica, los dispositivos suelen enlazarse mediante una pasarela y en la Figura 5 en la página 34(d) se muestran tres ejemplos de este tipo. Se prefiere una pasarela a un nodo de enlace porque las pasarelas soportan varias peticiones de conexión recibidas al mismo tiempo. La primera configuración que se muestra es la más probable, aunque la segunda y la tercera pueden realizarse, a pesar de que resulten poco frecuentes. Observe que en la tercera configuración sólo puede configurarse una ruta para un gestor de colas remoto determinado y, por lo tanto, aunque parezca que existan dos rutas, debe seleccionarse una.

En las redes de mayor tamaño se pueden utilizar varias pasarelas, cada una de las cuales dará soporte a varios dispositivos. Las pasarelas pueden conectarse entre sí de cualquier modo, pero si se define una interconexión completa entonces ninguna ruta que pase entre los dispositivos requerirá más de dos pasarelas. La Figura 6 muestra un ejemplo de una red de mayor tamaño.

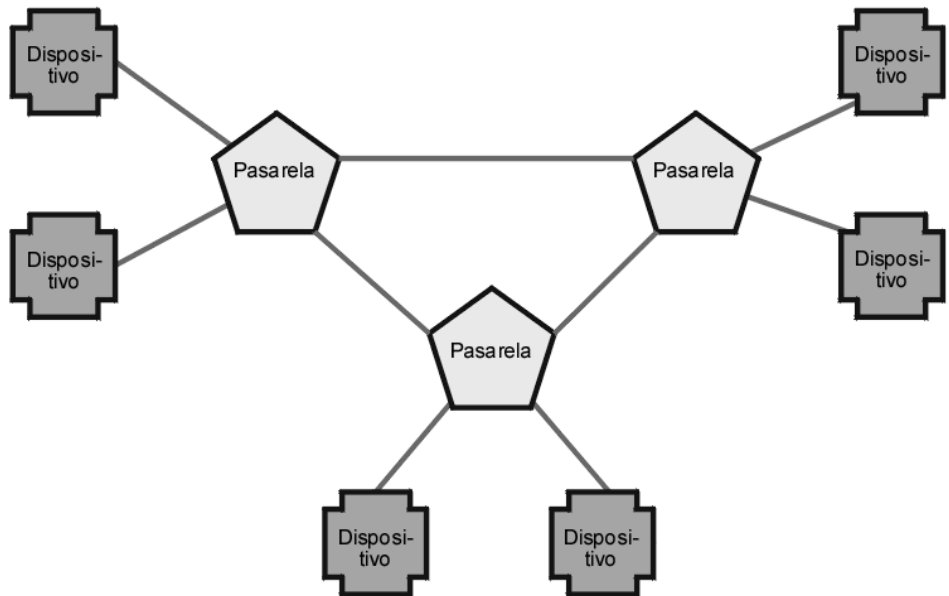


Figura 6. Una red MQSeries Everyplace en estrella

## 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 corresponde, 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 remitir 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 de MQSeries. Este mecanismo significa que es posible llevar a cabo el tipo de mensajería síncrona 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 a nivel de mensajes - protección a nivel de mensajes cuando los mensajes están entre el iniciador y el receptor

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

Las tres características de seguridad de MQSeries Everyplace protegen los datos de los mensajes utilizando un atributo (MQeAttribute 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" para, por ejemplo, descomprimir y descifrar los datos del mensaje).

Los algoritmos que soporta MQSeries Everyplace Versión 1.0 para autenticación, cifrado y compresión se describen detalladamente en la Tabla 13 en la página 38.

Tabla 13. Soporte de autenticación, cifrado y compresión

Función	Algoritmo
Autenticación	Basada en minicertificados (derivados de los minicertificados WTLS del foro WAP)
	Validación de identidad de Windows NT/2000
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) y asociando explícitamente la clave al atributo y, a continuación, asociando el atributo 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 de 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 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 mensajería asíncrona requiere una definición de cola remota, de lo contrario, las características de la cola de destino no se pueden determinar.

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.

Al igual que 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", 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", 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`. MQeMAttribute 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 una mayor optimización del tamaño, los certificados que se utilizan son minicertificados basados en certificados WTLS propuestos por la especificación WTLS del foro WAP. 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.

## 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 remoto
- 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 el 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 establecido para

utilizar autenticadores basados en minicertificados, como entidades autenticables. Por lo tanto, los gestores de colas que dan soporte a estos servicios pueden 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 minicertificados.

## **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 de 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 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 minicertificados 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 minicertificados (basados en minicertificados WTLS del foro WAP)
- Gestión del depósito de minicertificados

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 (en base a 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

---

## Configuración y 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, efectúan 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 normas cuando el código puente entre MQSeries Everyplace y 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.
- Los objetos administrados por el puente entre estados deberían iniciarse cuando se inicia la instancia del servidor.
- Cuando el puente encuentra que algo no va bien en la cola de sincronismo (Sync Q) 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 MQ, y viceversa. Los transformadores para efectuar la conversión entre mensajes MQSeries Everyplace y MQ 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 del marcador RAS**

Se pasa el control a esta clase de norma cuando el marcador RAS ha sufrido un cambio de estado para que indique qué se ha de hacer cuando, por ejemplo:

- El número al que se llama no se conecta
- Se ha sobrepasado un umbral de errores
- Se ha intentado marcar y sólo deben utilizarse determinados tipos de conexión, basados en la hora del día. Por ejemplo, cuando sólo se puede utilizar el teléfono fuera de las horas punta.

### **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 temporalización.
- 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 dar soporte a los tipos de operaciones de cliente/servidor<sup>10</sup> y de igual a igual. Un *cliente* puede iniciar las comunicaciones con un servidor; un *servidor* sólo puede responder a las peticiones iniciadas por 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.

La Tabla 14 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 de la servlet describe el caso en que se configura MQSeries Everyplace como una 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 14. 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
Servlet	Sí	Sí	

***El uso de un gestor de canales de MQSeries Everyplace o de un escucha de canal de MQSeries Everyplace determina, por razones de licencia, que una instancia de MQSeries Everyplace es una pasarela.***

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 en un dispositivo o pasarela y qué conexiones pueden existir al mismo tiempo.

---

10. En este apartado, los términos cliente y servidor reflejan el uso que se les da generalmente, no la semántica propiamente dicha de MQSeries.

## 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 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 puertas 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 15 en la página 48

en la página 48 resume las posibilidades. Se pueden identificar las clases de forma explícita o utilizando los nombres de alias.

*Tabla 15. Opciones de clases*

<b>Clase</b>	<b>Alternativas suministradas</b>	<b>Interfaces documentadas</b>
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 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 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, que 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 de MQSeries se pueden considerar simplemente como gestores de colas remotos 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 remoto) 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 puentes opcionales entre MQSeries Everyplace y otros sistemas de mensajería.

En MQSeries Everyplace Versión 1.0 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 MQSeries Everyplace Versión 1 se recomienda un puente de este tipo por pasarela, 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 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.

Un servicio de conexión de cliente de pasarela puede tener varios *escuchas* que utilicen este servicio de 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.

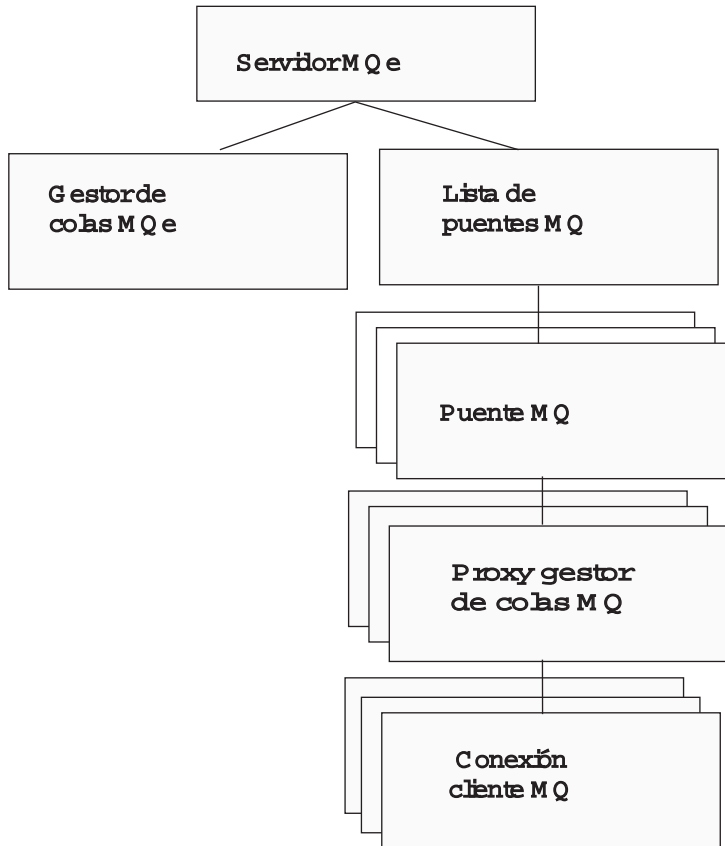


Figura 7. Jerarquía de los objetos del puente MQSeries

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 16.

*Tabla 16. Propiedades del objeto de puente*

Propiedad	Descripción
Nombre de puente	Lista del 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 17 muestra las propiedades del objeto de puente.

*Tabla 17. 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 a las 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 18 en la página 52.

*Tabla 18. Propiedades del proxy del gestor de colas MQSeries*

<b>Propiedad</b>	<b>Descripción</b>
Clase	La clase del proxy del gestor de colas MQSeries
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 cliente del puente contiene la información detallada que se necesita para realizar una conexión con un gestor de colas MQSeries. En la Tabla 19 se muestran las propiedades del servicio de conexión.

*Tabla 19. Propiedades del servicio de conexión del cliente*

<b>Propiedad</b>	<b>Descripción</b>
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 19. 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 segura. Después de una conclusión normal, la cola está vacía. La pueden compartir 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.

El objeto de escucha, que desplaza los mensajes de MQSeries a MQSeries Everyplace, tiene las propiedades que se muestran en la Tabla 20.

Tabla 20. 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	El 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.

Para completar la configuración del puente, son necesarias tanto las definiciones del gestor de colas remoto como las de la cola remota. Las definiciones de los gestores de colas remotos de MQSeries Everyplace siguen el mismo método estándar de MQSeries Everyplace. Las definiciones de los gestores de colas remotos de MQSeries tienen una definición de canal establecida en un valor nulo para indicar que no se utiliza un canal dinámico MQSeries Everyplace normal, y en su lugar, se define una conexión con el gestor de colas MQSeries como se ha descrito detalladamente anteriormente.

La definición de la una cola remota de MQSeries Everyplace sigue también el método estándar, sin embargo, en el caso de una cola MQSeries las diferencias son importantes en relación con las colas MQSeries Everyplace. La Tabla 21 en la página 55

página 55 muestra las propiedades de las colas remotas de MQSeries.

*Tabla 21. Propiedades de las colas remotas de MQSeries*

Propiedad	Descripción
Nombres de alias	Los nombres alternativos de la cola
Autenticador	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

Las clases del *cifrador*, del *autenticador* 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 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 MQSeries únicamente son síncronas. Por lo tanto, las aplicaciones asíncronas deben enviar los mensajes a estas colas a través de las colas para almacenar y remitir y de las colas del servidor inicial de MQSeries Everyplace, o a través de las definiciones de colas remotas.

La administración de la pasarela 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. La Tabla 11 en la página 29 muestra 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 MQS:**

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 MQ. 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 MQS

- 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 segura de una sola vez
- **Aplicaciones de ejemplo:** Interoperabilidad con las aplicaciones MQSeries Postales y MQSeries Everyplace Postales. <sup>11</sup>

MQSeries Everyplace Versión 1 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 subclasses, sustituyendo las clases suministradas o utilizando las normas, interfaces y otras características de personalización que incorpora el producto.

- No se da soporte a los clusters
- 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 incorpora activación
- No hay soporte de unidades de trabajo, ni coordinación de tipo XA

Las características de rendimiento y posibilidad de crecimiento son diferentes.

---

## Entrega segura

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 “Operaciones de los gestores de colas” en la página 26). Cuando un mensaje se desplaza desde MQSeries a MQSeries Everyplace, la transferencia únicamente se asegura si el mensaje MQSeries se ha definido como permanente.

---

11. Esta aplicación no se incluye en la Versión 1.0 pero se puede bajar del sitio web de software de MQSeries en la World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>).

---

## Capítulo 7. Aplicaciones y programas de utilidad

**Nota:** Estas aplicaciones y programas de utilidad no se incluyen en la Versión 1.0 pero se podrán bajar del sitio de software de MQSeries en la World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>).

---

### Postales

Postales es una aplicación MQSeries Everyplace que se puede utilizar para validar la operación de una red MQSeries Everyplace autónoma o las operaciones entre redes MQSeries Everyplace y MQSeries. Postales es una aplicación Java que permite enviar mensajes de texto a un usuario de un gestor de colas remoto. Puede operar con la aplicación Postales, de tal modo, que si se envía un mensaje de Postales a un destino MQSeries, se recibirá una postal, y viceversa.

También hay una versión C de la aplicación MQSeries Everyplace Postales que se ejecuta en los sistemas operativos Palm OS y puede operar con la versión Java.

---

### MQSeries Everyplace Explorer

MQSeries Everyplace Explorer es una herramienta de gestión, escrita en Java, que permite configurar y utilizar los gestores de colas, las colas y los mensajes locales y remotos. Utiliza las clases basadas en Microsoft® para presentar una interfaz gráfica de usuario estándar de Windows pero, como resultado, no se puede ejecutar en plataformas que no sean Windows. Sin embargo, se puede utilizar para gestionar todos los gestores de colas MQSeries Everyplace ya que puede operar exclusivamente a través de los mensajes de administración de envío y recepción. Presenta una vista de dos paneles de una red MQSeries Everyplace, una vista de árbol de los objetos en el panel de la izquierda y una vista de lista de detalles de objetos en el panel de la derecha.

MQSeries Everyplace Explorer puede realizar lo siguiente:

- Visualizar o modificar las propiedades del gestor de colas
- Crear, eliminar o modificar las conexiones y visualizar sus propiedades
- Crear, eliminar o modificar las colas y visualizar sus propiedades y su contenido
- Examinar o eliminar mensajes, visualizar sus propiedades e inspeccionar sus campos
- Enviar mensajes de prueba
- Configurar el puente entre MQSeries Everyplace y MQSeries

Normalmente, MQSeries Everyplace Explorer utiliza un gestor de colas que ya está configurado y puede cargar otras clases para su ejecución. Si no existe un gestor de colas de este tipo, esta herramienta creará uno con las características seleccionadas por el usuario.

Se pueden ejecutar varias copias de la herramienta en una sola máquina y cada una de ellas puede ejecutarse en su propia JVM. Esta disposición permite realizar una

simulación de una red MQSeries Everyplace que se puede utilizar para investigar y demostrar las funciones de red y las operaciones de MQSeries Everyplace.



---

## Capítulo 8. Interfaces de programación

*MQSeries Everyplace Systems Programming Interface (SPI)* es la interfaz de programación de MQSeries Everyplace. Existen dos implementaciones, dependiendo del sistema operativo. La versión Java proporciona acceso a todas las funciones de MQSeries Everyplace, la interfaz C de MQSeries Everyplace Versión 1.0 sólo proporciona acceso a un subconjunto. En la publicación *MQSeries Everyplace Programming Reference* se proporciona información detallada sobre las clases, métodos y procedimientos, en la publicación *Guía de programación de MQSeries Everyplace* se incluyen ejemplos de programación de MQSeries Everyplace.



---

## 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 “COMO ESTÉ” 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.

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  
Inglaterra  
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 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. Un método público o variable 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 colas 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 Léame primero*, GC10-3547-00
- *MQSeries Everyplace Programming Reference*, SC34-5846-00
- *Guía de programación de MQSeries Everyplace*, SC10-3548-00
- *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 ix  
acciones de rastreo en MQSeries Everyplace 33  
acerca de este manual ix  
adaptadores MQSeries Everyplace 32  
administración con MQSeries Everyplace 28  
anotaciones de sucesos 33  
aplicación Postales 59  
aplicaciones 59  
aplicaciones, carga de 48  
aplicaciones MQSeries Everyplace 11, 59  
asíncrona, mensajería 25  
autenticables, entidades 41  
avisos legales 63

## B

basada en colas, seguridad 38

## C

canales, de cliente 15  
canales, dinámicos 15, 31  
canales de cliente 7, 15  
canales dinámicos 7, 15, 31  
capacidad de crecimiento 34  
carga de las aplicaciones 48  
certificados, reproducción 42  
cifrado 37  
clases de MQSeries Everyplace 47  
cliente, canales 7  
cliente MQSeries 6  
cliente/servidor, conexión 47  
colas de MQSeries Everyplace 20  
colas del servidor inicial 22  
colas locales 21  
colas para almacenar y remitir 21  
colas puente de MQSeries 22  
colas remotas 21  
compatibilidad con MQSeries 57  
compresión 37  
comunicaciones 46  
conceptos del producto 15  
conexión cliente/servidor 47  
conexión de igual a igual 47  
configuración 44  
configuraciones de MQSeries Everyplace 34  
conocimientos previos necesarios ix  
conversión de mensajes 56

## D

de igual a igual, conexión 47  
descripción 1  
diferentes, tipos de conexiones 47  
dinámicos, canales 7

dispositivos de MQSeries Everyplace 15  
dispositivos MQSeries Everyplace 7

## E

entidades autenticables 41  
entrega de mensajes segura 58  
escucha de canal 46

## F

formato de los datos de las operaciones de vuelco 20

## G

gestión de conexiones de marcación 33  
gestor de canales 46  
gestor de colas 46  
gestores de colas 6  
gestores de colas, objetos 26  
gestores de colas de MQSeries Everyplace 24

## I

interfaces de programación 61  
interfaz con MQSeries 49  
interfaz de seguridad 44

## L

la familia MQSeries 5  
lectores ix  
legales, avisos 63  
local, seguridad 38  
locales, colas 21

## M

marcas registradas 64  
mensaje, objetos 16  
mensajería, asíncrona 35  
mensajería, síncrona 36  
mensajería asíncrona 25, 35  
mensajería de estación de trabajo 6  
mensajería de sistema principal 6  
mensajería distribuida 5  
mensajería general 6  
mensajería MQSeries 5  
mensajería síncrona 25, 36  
mensajes, de administración 28  
mensajes de administración 28  
minicertificados 41  
MQeAttribute 40  
MQeMTrustAttribute 40  
MQS, puente 49  
MQS, redes 49  
MQSeries, cliente 6  
MQSeries, compatibilidad con 57  
MQSeries, interfaz con 49

- MQSeries, mensajería 5
- MQSeries, servidor 6
- MQSeries Everyplace, adaptadores 32
- MQSeries Everyplace, aplicaciones 11
- MQSeries Everyplace, clases 47
- MQSeries Everyplace, colas 20
- MQSeries Everyplace, configuraciones 34
- MQSeries Everyplace, dispositivos 7, 15
- MQSeries Everyplace, gestores de colas 24
- MQSeries Everyplace, normas 44
- MQSeries Everyplace, objetos 16
- MQSeries Everyplace, pasarelas 15
- MQSeries Everyplace, redes 33, 49
- MQSeries Everyplace, registro 15
- MQSeries Everyplace, seguridad 37
- MQSeries Everyplace, servidor inicial 22
- MQSeries Everyplace Explorer, programa de utilidad 59
- MQSeries Integrator 5
- MQSeries Workflow 5

## N

- nivel de mensajes, seguridad 40
- normas de atributos 44
- normas de colas 45
- normas de MQSeries Everyplace 44
- normas de puente de MQSeries 44
- normas del gestor de colas 45
- normas del marcador RAS 45

## O

- objeto de escucha 54, 55
- objeto de puente 51
- objeto del proxy del gestor de colas 51
- objetos de mensaje 16
- objetos de MQSeries Everyplace 16
- operaciones de los gestores de colas 26

## P

- para almacenar y remitir, colas 21
- pasarelas, MQSeries Everyplace 7
- pasarelas MQSeries Everyplace 7, 15
- personalización 44
- posibilidades 11
- Postales, aplicación 59
- privado, registro 42
- producto, conceptos 15
- programa de utilidad de administración 59
- programa de utilidad de administración MQSeries Everyplace 59
- programa de utilidad MQSeries Everyplace Explorer 59
- programación, interfaces 61
- programas de utilidad 59
- programas de utilidad MQSeries Everyplace 59
- público, registro 42
- puente de MQSeries, colas 22

- puente MQSeries 49

## R

- redes MQSeries 49
- redes MQSeries Everyplace 33, 49
- registro 41
- registro automático 42
- registro de MQSeries Everyplace 15, 41
- registro privado 42
- registro público 42
- remotas, colas 21
- reproducción de certificados 42
- requisitos de los usuarios 12
- requisitos previos 3

## S

- segura, entrega de mensajes 58
- seguridad, interfaz 44
- seguridad a nivel de mensajes 40
- seguridad basada en colas 38
- seguridad en MQSeries Everyplace 37
- seguridad local 38
- servicio de emisión de minicertificados 43
- servidor MQSeries 6
- síncrona, mensajería 25
- sistemas operativos necesarios 3
- sistemas operativos soportados 3
- soportados, sistemas operativos 3
- SPI 61
- supervisión 31

## T

- términos ix
- tipos de conexiones 46
- tipos de conexiones diferentes 47
- transformadores 56

## U

- usuarios, requisitos 12

## V

- visión general 1





Printed in Denmark by IBM Danmark A/S

GC10-3546-00

