

# **Sun GlassFish Message Queue 4.1 Update 1 - Notas de la versión**



Referencia: 821-1512-10  
Diciembre de 2009

Copyright ©2009 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Reservados todos los derechos.

Sun Microsystems, Inc. tiene derechos de propiedad intelectual relacionados con la tecnología del producto que se describe en este documento. En concreto, y sin limitarse a ello, estos derechos de propiedad intelectual pueden incluir una o más patentes de EE. UU. o aplicaciones pendientes de patente en EE. UU. y otros países.

Derechos del gobierno de los EE. UU. – Software comercial. Los usuarios de instituciones gubernamentales están sujetos al acuerdo de licencia estándar de Sun Microsystems, Inc. y a las disposiciones aplicables de FAR y de sus suplementos.

Esta distribución puede incluir componentes desarrollados por terceros.

Determinadas partes del producto pueden derivarse de Berkeley BSD Systems, con licencia de la Universidad de California. UNIX es una marca registrada en los EE. UU. y otros países, bajo licencia exclusiva de X/Open Company, Ltd.

Sun, Sun Microsystems, el logotipo de Sun, el logotipo de Solaris, el logotipo de la taza de café de Java, docs.sun.com, Java y Solaris son marcas comerciales o marcas comerciales registradas de Sun Microsystems, Inc. en EE. UU. y otros países. Todas las marcas registradas SPARC se usan con licencia y son marcas comerciales o marcas registradas de SPARC International, Inc. en los EE. UU. y en otros países. Los productos con las marcas registradas de SPARC se basan en una arquitectura desarrollada por Sun Microsystems, Inc.

La interfaz gráfica de usuario OPEN LOOK y Sun™ fue desarrollada por Sun Microsystems, Inc. para sus usuarios y licenciatarios. Sun reconoce los esfuerzos pioneros de Xerox en la investigación y el desarrollo del concepto de interfaces gráficas o visuales de usuario para el sector informático. Sun dispone de una licencia no exclusiva de Xerox para la interfaz gráfica de usuario de Xerox, que también cubre a los licenciatarios de Sun que utilicen las interfaces gráficas OPEN LOOK y que, por otra parte, cumplan con los acuerdos de licencia por escrito de Sun.

Los productos que se tratan y la información contenida en esta publicación están controlados por las leyes de control de exportación de los Estados Unidos y pueden estar sujetos a leyes de exportación o importación en otros países. Queda terminantemente prohibido el uso final (directo o indirecto) de esta documentación para el desarrollo de armas nucleares, químicas, biológicas, de uso marítimo nuclear o misiles. Queda terminantemente prohibida la exportación o reexportación a países sujetos al embargo de los Estados Unidos o a entidades identificadas en las listas de exclusión de exportación de los Estados Unidos, incluidas, aunque sin limitarse a ellas, las personas con acceso denegado y las listas de ciudadanos designados con carácter especial.

ESTA DOCUMENTACIÓN SE PROPORCIONA "TAL CUAL" Y SE RENUNCIA A TODAS LAS CONDICIONES EXPRESAS O IMPLÍCITAS, REPRESENTACIONES Y GARANTÍAS, INCLUIDAS CUALQUIER GARANTÍA IMPLÍCITA DE COMERCIALIZACIÓN, ADECUACIÓN PARA UNA FINALIDAD DETERMINADA O DE NO CONTRAVENCIÓN, EXCEPTO EN AQUELLOS CASOS EN QUE DICHA RENUNCIA NO FUERA LEGALMENTE VÁLIDA.

---

Copyright ©2009 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuelle relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs brevets américains ou des applications de brevet en attente aux Etats-Unis et dans d'autres pays.

Cette distribution peut comprendre des composants développés par des tierces personnes.

Certains composants de ce produit peuvent être dérivés du logiciel Berkeley BSD, licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays; elle est licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, le logo Solaris, le logo Java Coffee Cup, docs.sun.com, Java et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc., ou ses filiales, aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

Les produits qui font l'objet de cette publication et les informations qu'il contient sont régis par la législation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes chimiques ou biologiques ou pour le nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des Etats-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la législation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement désignés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

# Contenido

---

<b>1</b>	<b>Notas de la versión</b>	5
	Historial de revisiones de las notas de la versión	6
	Acerca de Message Queue 4.4 Update 1	7
	Plataformas y componentes compatibles con Message Queue 4.4 Update 1	7
	Sistemas operativos compatibles	7
	Virtualización del sistema compatible	8
	Dependencias de componentes	8
	Nuevas funciones de Message Queue 4.4 Update 1 y versiones recientes	10
	Nuevas funciones de Message Queue 4.4 Update 1	10
	Nuevas funciones de Message Queue 4.4	11
	Nuevas funciones de Message Queue 4.3	13
	Nuevas funciones de Message Queue 4.2	19
	Nuevas funciones de Message Queue 4.1	22
	Nuevas funciones de Message Queue 4.0	26
	Funciones que se eliminarán en futuras versiones	30
	Errores solucionados en Message Queue 4.4 Update 1 y versiones recientes	31
	Errores solucionados en Message Queue 4.4 Update 1	31
	Errores solucionados en Message Queue 4.4	33
	Errores solucionados en Message Queue 4.3	36
	Errores solucionados en Message Queue 4.2	36
	Errores solucionados en Message Queue 4.1	38
	Errores solucionados en Message Queue 4.0	39
	Actualizaciones de documentación de Message Queue 4.4 Update 1	40
	Problemas de compatibilidad	40
	Cambios en el conjunto de documentación de Message Queue 4.4 Update 1	41
	Limitaciones y problemas conocidos	42
	Problemas relacionados con la instalación	42
	Opción de contraseña eliminada	48

Problemas de administración y configuración .....	49
Problemas de agentes .....	51
Agrupaciones de agentes .....	52
Problemas con JMX .....	54
Compatibilidad con SOAP .....	55
Archivos redistribuibles .....	55
Funciones de accesibilidad para personas con discapacidad .....	55
Cómo informar sobre problemas y enviar comentarios .....	56
Foro de Sun GlassFish .....	56
Foro de tecnología Java .....	56
Sun agradece sus comentarios .....	57
Recursos adicionales de Sun .....	57

# Notas de la versión

---

## Versión 4.4 Update 1

Número de referencia 821-1512-10

Estas notas de la versión contienen información importante que está disponible en el momento del lanzamiento de Sun GlassFish Message Queue 4.4 Update 1. Además de tratarse otros temas, aquí se describen las funciones, las mejoras, las limitaciones y los problemas conocidos del producto. Antes de empezar a utilizar Message Queue 4.4 Update 1 lea este documento.

Estas notas de la versión también contienen información sobre las versiones 4.4, 4.3, 4.2, 4.1 y 4.0 de Message Queue. Por ejemplo, consulte “[Nuevas funciones de Message Queue 4.2](#)” en la página 19, “[Nuevas funciones de Message Queue 4.1](#)” en la página 22 y “[Nuevas funciones de Message Queue 4.0](#)” en la página 26, respectivamente, para obtener información sobre las nuevas funciones de cada versión.

La versión más actualizada de estas notas de la versión se encuentra en el sitio web de documentación de Sun GlassFish Message Queue, <http://docs.sun.com/coll/1307.7>. Consulte el sitio web antes de instalar y configurar el software y, después, visítelo de forma periódica para ver los manuales y las notas de la versión más actualizados.

En estas notas de la versión se incluyen los siguientes apartados:

- “Historial de revisiones de las notas de la versión” en la página 6
- “Acerca de Message Queue 4.4 Update 1” en la página 7
- “Plataformas y componentes compatibles con Message Queue 4.4 Update 1” en la página 7
- “Nuevas funciones de Message Queue 4.4 Update 1 y versiones recientes” en la página 10
- “Funciones que se eliminarán en futuras versiones” en la página 30
- “Errores solucionados en Message Queue 4.4 Update 1 y versiones recientes” en la página 31
- “Actualizaciones de documentación de Message Queue 4.4 Update 1” en la página 40
- “Limitaciones y problemas conocidos” en la página 42
- “Archivos redistribuibles” en la página 55
- “Funciones de accesibilidad para personas con discapacidad” en la página 55
- “Cómo informar sobre problemas y enviar comentarios” en la página 56

- [“Sun agradece sus comentarios” en la página 57](#)
- [“Recursos adicionales de Sun” en la página 57](#)

Este documento contiene referencias a direcciones URL de otras organizaciones para proporcionar información adicional relacionada.

Sun no se responsabiliza de la disponibilidad de las páginas web de otras empresas que se mencionan en este documento. Sun no avala ni se hace responsable del contenido, la publicidad, los productos ni otros materiales disponibles en dichos sitios o recursos, o a través de ellos. Sun tampoco se hace responsable de los daños o pérdidas, supuestos o reales, provocados por el uso o la confianza puesta en el contenido, los bienes o los servicios disponibles en dichos sitios o recursos, o a través de ellos.

## Historial de revisiones de las notas de la versión

La siguiente tabla contiene una lista de las fechas de todas las versiones 4.x del producto Message Queue y describe los cambios realizados en este documento en relación a cada versión.

**TABLA 1-1** Historial de revisiones

Fecha	Descripción de los cambios
Diciembre de 2009	Publicación de este documento para Message Queue 4.4 Update 1. Añade nuevas funciones de esta versión y elimina problemas de instalación obsoletos del programa de instalación anterior.
Diciembre de 2009	Segunda versión de este documento para Message Queue 4.4. Corrige errores y omisiones.
Octubre de 2009	Versión de este documento para Message Queue 4.4. Incluye nuevas funciones de esta versión.
Mayo de 2009	Versión inicial de este documento para Message Queue 4.4 Beta. Incluye nuevas funciones de esta versión.
Diciembre de 2008	Versión de este documento para Message Queue 4.3. Incluye nuevas funciones de esta versión.
Agosto de 2008	Versión de este documento para Message Queue 4.2. Incluye nuevas funciones de esta versión.
Septiembre de 2007	Tercera versión de este documento para Message Queue 4.1. Incluye la descripción de la compatibilidad con Java Enterprise System Monitoring Framework, puertos C fijos, errores corregidos y otras funciones.
Abril de 2007	Segunda versión de este documento para Message Queue 4.1 Beta. Incorpora la función de alta disponibilidad.
Enero de 2007	Versión inicial de este documento para Message Queue 4.1 Beta. Incorpora la descripción de la compatibilidad con JAAS.

TABLA 1-1 Historial de revisiones (Continuación)

Fecha	Descripción de los cambios
Mayo de 2006	Versión inicial de este documento para Message Queue 4.0.

## Acerca de Message Queue 4.4 Update 1

Sun GlassFish Message Queue es un servicio de mensajería completo que proporciona funciones fiables y asíncronas, conformes con la especificación Java Messaging Specification (JMS) 1.1. Además, Message Queue incluye funciones que van más allá de la especificación JMS para satisfacer las necesidades de las instalaciones en grandes empresas.

Message Queue 4.4 Update 1 es una versión secundaria que proporciona un nuevo programa de instalación basado en el sistema de empaquetado de imagen IPS independiente de la plataforma pkg (5). Además, versión 4.4 Update 1 incluye una serie de mejoras en las funciones y soluciones de errores.

## Plataformas y componentes compatibles con Message Queue 4.4 Update 1

Esta sección cubre los siguientes temas relacionados con los requisitos del sistema para Message Queue 4.4 Update 1:

- “Sistemas operativos compatibles” en la página 7
- “Virtualización del sistema compatible” en la página 8
- “Dependencias de componentes” en la página 8

## Sistemas operativos compatibles

Message Queue 4.4 Update 1 es compatible con sistemas operativos Solaris, Linux, Windows y AIX. En la [Tabla 1-2](#) se muestran las versiones compatibles de cada sistema operativo. Para obtener información sobre los requisitos de hardware de cada plataforma, consulte la *Sun Java System Message Queue 4.3 Installation Guide*

TABLA 1-2 Versiones de sistemas operativos compatibles

Plataforma	Versiones compatibles
Solaris	Solaris 9 (SunOS 5.9), todas las actualizaciones (SPARC, x86)
	Solaris 10 (SunOS 5.10), todas las actualizaciones (SPARC, x86, x64)
OpenSolaris	OpenSolaris 2008.11 (x86, x64)
	OpenSolaris 2009.06 (x86, x64)

TABLA 1-2 Versiones de sistemas operativos compatibles (Continuación)

Plataforma	Versiones compatibles
Linux	Red Hat Enterprise Linux Advanced Server 4.0, 5.0, todas las actualizaciones, versiones de 32 y 64 bits (x86, x64)  Red Hat Enterprise Linux Enterprise Server 4.0, 5.0, todas las actualizaciones, versiones de 32 y 64 bits (x86, x64)
AIX	AIX 6.1
Windows	Windows Vista  Windows XP Professional, SP2 ( x86) <sup>1</sup>  Windows Server 2003 Standard Edition y Enterprise Edition, SP2, versiones de 32 y 64 bits (x86, x64) <sup>2</sup>  Windows Server 2008 Standard Edition y Enterprise Edition, SP2, versiones de 32 y 64 bits (x86, x64) <sup>2</sup>

<sup>1</sup> No es compatible con Home, Tablet PC ni Media Center Edition

<sup>2</sup> No es compatible con Web ni Small Business Server Edition

## Virtualización del sistema compatible

La virtualización del sistema es una tecnología que permite que varias instancias del sistema operativo (SO) se ejecuten de forma independiente en un hardware compartido. Desde el punto de vista de la funcionalidad, el software ejecutado en un SO alojado en un entorno virtualizado no detecta, normalmente, que la plataforma subyacente se ha virtualizado. Sun realiza pruebas de sus productos de Sun Java System en determinadas combinaciones de SO y virtualización de sistemas para confirmar que los productos de Sun Java System siguen funcionando en entornos virtualizados con una configuración y un tamaño correctos, del mismo modo que lo harían en sistemas que no se hayan virtualizado. Para obtener información sobre los productos Sun Java System compatibles con entornos virtualizados, consulte <http://docs.sun.com/doc/820-4651>.

## Dependencias de componentes

Además de los requisitos específicos de cada plataforma, Message Queue también depende de determinados componentes básicos que deben instalarse para desarrollar y ejecutar clientes Message Queue. En la [Tabla 1-3](#) se describen estos componentes. Asimismo, pueden utilizarse versiones distintas o implementaciones de otros proveedores, pero Sun Microsystems no las ha probado y, por tanto, no garantiza oficialmente su compatibilidad.

TABLA 1-3 Componentes necesarios

Componente	Admite	Versiones compatibles <sup>1</sup>
Java Runtime Environment (JRE)	Herramientas para agentes y administración de Message Queue	Entorno de tiempo de ejecución J2SE™ 1.5.0 o posterior Entorno de tiempo de ejecución de Java™ SE 1.6.0
Java Software Development Kit (JDK), Standard Edition	Desarrollo e implementación de clientes Java	Kit de desarrollo J2SE™ 1.5.0 o posterior Kit de desarrollo Java SE 1.6.0. Tenga en cuenta que JMX requiere la versión 1.6.0_10 o posterior.

<sup>1</sup> Sólo versiones de producción de Sun Microsystems

En la [Tabla 1-4](#) se muestran los componentes adicionales que puede instalar para dar mayor soporte a los clientes Message Queue. Es posible que no necesite todos los componentes de la lista, por ejemplo, si no piensa escribir en C, no necesitará el compilador C, la biblioteca de tiempo de ejecución C++, NSPR ni NSS.

TABLA 1-4 Componentes de compatibilidad opcionales

Componente	Admite	Versiones compatibles
Servidor de aplicaciones	HTTP/HTTPS	Application Server Enterprise Edition, Version 9.1.1 de Sun GlassFish Sun GlassFish Enterprise Server v2.1, v2.1.1, v3
Servidor web	HTTP/HTTPS	Sun GlassFish Web Server Enterprise Edition, Versión 7.0, Update 3
Base de datos	Almacén de datos basado en JDBC	HADB, versiones 4.4.3, 4.5, 4.6 Java DB (Apache Derby), Versión 10.4 MySQL Community/Enterprise Edition, Versión 5.1 Oracle 9.2i, 10g y 11g postgresql, Versión 8.1 <b>Nota</b> – La base de datos PointBase ya no es compatible.
Base de datos de alta disponibilidad	Clústeres de agente de alta disponibilidad	HADB, versiones 4.4.3, 4.5, 4.6 MySQL Cluster Edition, Versión 5.1.39-ndb-7.0.9 Oracle 10g y 11g

TABLA 1-4 Componentes de compatibilidad opcionales (Continuación)

Componente	Admite	Versiones compatibles
Servidor de directorios Lightweight Directory Access Protocol (LDAP)	Objetos administrados y almacén del usuario de Message Queue	Sun GlassFish Directory Server, Versión 6.0
Interfaz de directorios y nombres Java (JNDI)	Compatibilidad con objetos administrados y almacén del usuario LDAP	JNDI Versión 1.2.1 LDAP Service Provider, Versión 1.2.2 File System Service Provider, Versión 1.2 Beta 3 <sup>1</sup>
Compilador C y biblioteca de tiempo de ejecución C++ compatible	Clientes C de Message Queue	<b>Solaris:</b> Sun Studio, Versión 12 o posterior, compilador C++ con modo estándar y compilador C <b>Linux:</b> gcc/g++, Versión 3.4.6 <b>Windows:</b> Microsoft Windows Visual Studio, Versión 2008 SP1
Netscape Portable Runtime (NSPR)	Clientes C de Message Queue	Versión 4.7
Network Security Services (NSS)	Clientes C de Message Queue	Versión 3.12.3

<sup>1</sup> Sólo compatible con objetos administrados; permite desarrollar y comprobar el código, pero no implementar aplicaciones en un entorno de producción

## Nuevas funciones de Message Queue 4.4 Update 1 y versiones recientes

Las nuevas funciones de Message Queue 4.4 Update 1 y versiones anteriores de la familia Message Queue 4.x se describen en las siguientes secciones:

- [“Nuevas funciones de Message Queue 4.4 Update 1” en la página 10](#)
- [“Nuevas funciones de Message Queue 4.4” en la página 11](#)
- [“Nuevas funciones de Message Queue 4.3” en la página 13](#)
- [“Nuevas funciones de Message Queue 4.2” en la página 19](#)
- [“Nuevas funciones de Message Queue 4.1” en la página 22](#)
- [“Nuevas funciones de Message Queue 4.0” en la página 26](#)

## Nuevas funciones de Message Queue 4.4 Update 1

Message Queue 4.4 Update 1 es una versión menor que incluye nuevas mejoras en lo que a funciones y soluciones a fallos se refiere. En esta sección se describen las nuevas funciones incluidas en esta versión:

- “Nuevo programa de instalación” en la página 11
- “Admisión de registros de transacciones para agrupaciones” en la página 11
- “Agente interno del proceso” en la página 11

## Nuevo programa de instalación

Message Queue 4.4 Update 1 incluye un nuevo instalador multiplataforma basado en el sistema pkq (5), también conocido como IPS (Image Packaging System). Para obtener más información sobre este programa de instalación, consulte [Sun GlassFish Message Queue 4.4 Update 1 Installation Guide](#).

## Admisión de registros de transacciones para agrupaciones

Message Queue 4.4 Update 1 incluye un mecanismo de persistencia de transacción para almacenes de datos basados en archivos que es compatible con agrupaciones de agentes. Este mecanismo proporciona otras funciones, descritas en “[Optimizing File-Based Transaction Persistence](#)” de [Sun GlassFish Message Queue 4.4 Administration Guide](#).

## Agente interno del proceso

Message Queue 4.4 Update 1 permite ejecutar un agente desde el cliente Java. Un agente de este tipo, llamado agente *interno del proceso* o *incrustado*, se ejecuta en la misma JVM que el cliente Java encargado de la creación y el inicio. Si necesita más información, consulte el [Capítulo 6](#), “[Embedding a Message Queue Broker in a Java Client](#)” de [Sun GlassFish Message Queue 4.4 Developer’s Guide for Java Clients](#).

# Nuevas funciones de Message Queue 4.4

Message Queue 4.4 es una versión secundaria que incluye nuevas mejoras de funciones y soluciones de errores. En esta sección se describen las nuevas funciones incluidas en esta versión:

- “Servicio de conexión JMS” en la página 11
- “Servicio de conexión STOMP” en la página 12
- “Mejoras adicionales” en la página 13

## Servicio de conexión JMS

Debido a que la especificación JMS no define un protocolo para la comunicación entre agentes y clientes, cada proveedor de soluciones JMS (incluido Message Queue) utiliza su propio protocolo patentado. Esta situación ha provocado que las soluciones JMS de diferentes proveedores no sean compatibles.

El servicio de conexión JMS de Message Queue 4.4 soluciona este problema al permitir que un agente de Message Queue asigne sus destinos a destinos de proveedores JMS externos. Esta asignación permite que el agente de Message Queue se comunique de manera efectiva con clientes del proveedor JMS externo.

El servicio de conexión JMS permite asignar destinos de proveedores JMS externos que:

- Sean compatibles con JMS 1.1
- Admitan objetos administrados JNDI
- Utilicen fabricas de conexión de tipo `javax.jms.ConnectionFactory` o `javax.jms.XAConnectionFactory`
- Para la asignación con transacción, se admiten interfaces XA como administrador de recursos

Muchas soluciones JMS de otros proveedores, tanto de código abierto como comerciales, cumplen estos requisitos; por lo que el servicio de conexión JMS es un sistema efectivo para integrar Message Queue en entornos de mensajería con otros proveedores JMS.

Si necesita más información sobre el servicio de conexión JMS, consulte:

- Si necesita más información sobre la arquitectura, los subcomponentes y las funciones del servicio de conexión JMS, consulte la sección “[JMS Bridge Service](#)” de *Sun GlassFish Message Queue 4.4 Technical Overview*.
- Si necesita más información sobre cómo configurar y administrar las conexiones JMS de un agente, consulte la sección “[Configuring and Managing JMS Bridge Services](#)” de *Sun GlassFish Message Queue 4.4 Administration Guide*.

## Servicio de conexión STOMP

Como se ha mencionado anteriormente, la especificación JMS no define un protocolo de conexión para la comunicación entre agentes y clientes. El proyecto de código abierto STOMP (Streaming Text Oriented Messaging Protocol) de <http://stomp.codehaus.org> define un protocolo de conexión simple que pueden usar clientes escritos en cualquier lenguaje para comunicarse con cualquier proveedor de mensajería compatible con el protocolo STOMP.

Message Queue 4.4 es compatible con el protocolo STOMP mediante el servicio de conexión STOMP. Este servicio permite a un agente de Message Queue comunicarse con clientes STOMP.

Si necesita más información sobre el servicio de conexión STOMP, consulte:

- Si necesita más información sobre la arquitectura y las funciones del servicio de conexión STOMP, consulte la sección “[STOMP Bridge Service](#)” de *Sun GlassFish Message Queue 4.4 Technical Overview*.
- Si necesita más información sobre cómo configurar y administrar una conexión STOMP de un agente, consulte la sección “[Configuring and Managing STOMP Bridge Services](#)” de *Sun GlassFish Message Queue 4.4 Administration Guide*.

## Mejoras adicionales

Message Queue 4.4 también incluye las siguientes mejoras adicionales:

- “Nuevas funciones UMS (Universal Message Service)” en la página 13
- “Admisión de paquetes IPS” en la página 13
- “Función de registro de auditoría reincorporada” en la página 13

## Nuevas funciones UMS (Universal Message Service)

Ahora, el servicio UMS proporciona funciones que utilizan HTTP GET para ofrecer múltiples servicios:

- **getBrokerInfo**: obtiene información sobre el agente.
- **getConfiguración**: obtiene información sobre la configuración UMS.
- **debug**: activa o desactiva el registro de errores en el servidor UMS.
- **ping**: se comunica con el agente para confirmar que esté en ejecución.

Para obtener más información sobre estas nuevas funciones, consulte “[Query and utility functions using HTTP GET](https://mq.dev.java.net/4.4-content/imqums/protocol.html)” en <https://mq.dev.java.net/4.4-content/imqums/protocol.html>.

Para ver una descripción general de UMS, consulte “[Universal Message Service \(UMS\)](https://mq.dev.java.net/4.4-content/imqums/protocol.html)” en la página 14. Para consultar la documentación de la API UMS, visite <https://mq.dev.java.net/4.4-content/imqums/protocol.html>. Para ver ejemplos de programación en diferentes lenguajes, consulte <https://mq.dev.java.net/4.4-content/imqums/examples/README.html>.

## Admisión de paquetes IPS

Ahora, Message Queue se empaqueta para su distribución con el sistema de código abierto IPS (Image Packaging System), también conocido como sistema pkg (5). Este sistema de empaquetado se utiliza para integrar Message Queue con Sun GlassFish Enterprise Server 2.1.1.

## Función de registro de auditoría reincorporada

Message Queue 3.7 incluía una función de registro de auditoría que se eliminó en Message Queue 4.0. Esta función se ha reincorporado en Message Queue 4.4. Para obtener más información acerca de esta función, consulte “[Audit Logging](#)” de *Sun GlassFish Message Queue 4.4 Administration Guide*.

## Nuevas funciones de Message Queue 4.3

Message Queue 4.3 es una versión secundaria que incluye varias mejoras de funciones y errores corregidos. En esta sección se describen las nuevas funciones incluidas en esta versión:

- “[Universal Message Service \(UMS\)](#)” en la página 14

- “Compatibilidad con la plataforma AIX” en la página 16
- “Nuevo instalador basado en Zip” en la página 17
- “Compatibilidad con plataformas ampliada” en la página 17
- “Mejoras adicionales” en la página 17

## Universal Message Service (UMS)

Message Queue 4.3 incluye un nuevo servicio UMS (Universal Messaging Service) y una API de mensajería que permiten acceder a Message Queue desde cualquier dispositivo con http. Como resultado, prácticamente cualquier aplicación puede comunicarse con otras aplicaciones y aprovechar la fiabilidad y el envío garantizado de la mensajería JMS. Además, el servicio UMS ofrece una mayor escalabilidad para la mensajería JMS, lo que permite aumentar el número de clientes de mensajería hasta proporciones de escala de Internet.

## Arquitectura

La siguiente figura muestra la arquitectura UMS básica:

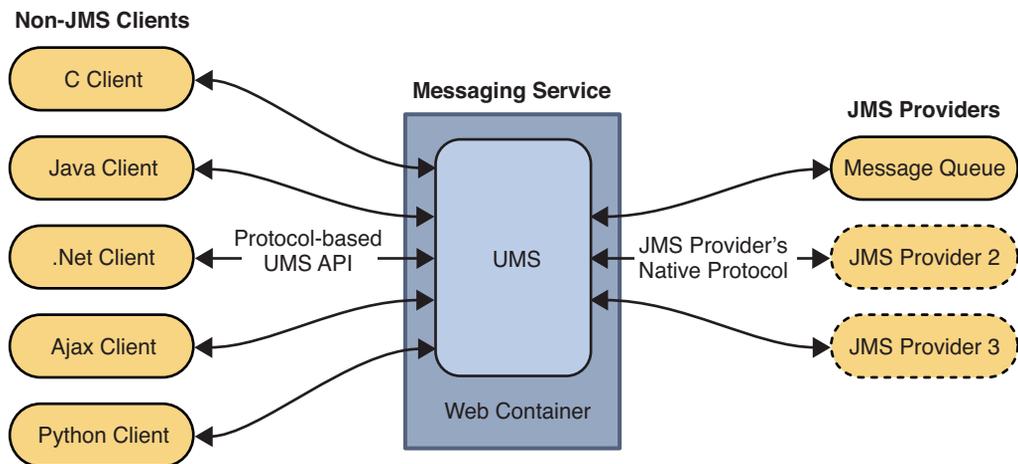


FIGURA 1-1 Arquitectura UMS

UMS, que se ejecuta en un servidor web, es un servicio independiente de la plataforma y con lenguaje neutro. El servicio UMS actúa como una puerta de enlace entre cualquier aplicación cliente no JMS y un proveedor JMS. Recibe mensajes enviados con la API UMS, los transforma en mensajes JMS y los produce en destinos del proveedor JMS mediante el protocolo nativo del proveedor. De modo similar, extrae mensajes de destinos del proveedor JMS, los transforma en mensajes de texto o SOAP y los envía a clientes no JMS según lo soliciten los clientes mediante la API UMS.

La API UMS, sencilla, independiente del lenguaje y basada en protocolo, admite tanto aplicaciones web como no basadas en web, y puede usarse con lenguajes de secuencias de comandos y de programación. La API está disponible en dos modelos: una API de mensajería sencilla que utiliza un protocolo de tipo REST (REpresentational State Transfer), y una API de mensajería XML que incrusta el protocolo en un encabezado de mensaje SOAP. En ambos modelos, la API sólo necesita una solicitud http para enviar o recibir un mensaje.

La sencillez y flexibilidad de la API UMS permite que aplicaciones AJAX, .NET, Python, C, Java y de otro tipo puedan enviar mensajes de texto o SOAP (con adjuntos) a destinos JMS o recibir mensajes de destinos JMS. Por ejemplo, las aplicaciones Python pueden comunicarse con aplicaciones .NET, las aplicaciones de iPhone pueden comunicarse con aplicaciones Java, etc.

Para Message Queue 4.3, el servicio UMS sólo admite Message Queue como proveedor JMS.

## Funciones adicionales

La función del servicio UMS va más allá que una simple puerta de enlace, descrita anteriormente. Admite sesiones cliente con estado y sin estado. Si lo solicita el cliente, el servicio UMS mantendrá el estado de sesión de la aplicación cliente en varias solicitudes de servicio. El servicio UMS puede utilizar autenticación administrada por contenedor, o configurarse para autenticar los clientes mediante el agente de Message Queue, o ambos. El servicio UMS también admite transacciones, lo que permite a las aplicaciones cliente confirmar o deshacer varias solicitudes de servicio como una única unidad atómica.

El servicio UMS admite un gran número de clientes en una única conexión con el agente de Message Queue, lo que reduce la carga de los servicios de conexión del agente y permite una escalabilidad máxima. Además, la capacidad del servicio UMS puede aumentarse con escalado horizontal, lo que permite trabajar con cargas de mensajes de escala de Internet.

En el cliente, gracias a la sencillez de la API UMS basada en protocolo, no se necesitan bibliotecas de cliente. Como resultado, la API puede ampliarse en el futuro para aplicar nuevas funciones JMS sin necesidad de actualizar las aplicaciones del cliente.

## Uso del servicio UMS

Para utilizar el servicio UMS debe implementar UMS en un contenedor web compatible con las especificaciones Servlet 2.4 o posterior, iniciar el agente de Message Queue, crear los destinos correspondientes y escribir una aplicación de mensajería que utilice la API UMS para enviar o recibir mensajes.

El archivo UMS `imqums.war`, incluido en Message Queue 4.3, se instala en la siguiente ubicación, según la plataforma:

Puede cambiar el nombre del archivo `.war` según sus necesidades.

TABLA 1-5 Ubicación del archivo `imqums.war`

Plataforma	Ubicación de <code>imqums.war</code>
Solaris	<code>/usr/share/lib/imq</code>
Linux	<code>/opt/sun/mq/share/lib</code>
AIX	<code>IMQ_HOME/lib</code>
Windows	<code>IMQ_HOME\lib</code>

Cuando haya implementado el archivo `imqums.war` en un contenedor web en `localhost:puerto`, podrá consultar la documentación de UMS en:

`http://localhost:puerto/imqums`

De lo contrario, puede encontrar la documentación de UMS como se indica a continuación:

- Para obtener información acerca de la configuración de UMS, consulte <https://mq.dev.java.net/4.3-content/ums/config.html>.
- Si necesita documentación sobre la API UMS, consulte <https://mq.dev.java.net/4.3-content/ums/protocol.html>.
- Para ver ejemplos de programación en diferentes lenguajes, consulte <https://mq.dev.java.net/4.3-content/ums/examples/README.html>.

## Contenedores web compatibles

Actualmente, los siguientes contenedores web admiten UMS:

- Sun GlassFish Enterprise Server, Versión 2.1 y Versión 3 Prelude
- Tomcat, versiones 5.5 y 6.0

## Compatibilidad con la plataforma AIX

Message Queue 4.3 incluye paquetes para la plataforma AIX y el instalador correspondiente.

La implementación AIX de Message Queue admite el siguiente software:

- AIX v 6.1 o posterior (se admiten versiones anteriores de AIX mediante el paquete Unix/Java Only)
- Admisión de DB2
- Compilador C/C++ IBM XL V9.0
- JDK 1.5 o superior

Si necesita instrucciones de instalación, consulte el Capítulo 4, “AIX Installation,” de la *Sun Java System Message Queue 4.3 Installation Guide*.

En la plataforma AIX, los archivos de Message Queue se instalan en un único directorio raíz de Message Queue, `IMQ_HOME`. `IMQ_HOME` se refiere al directorio `mqInstallHome/mq, mqInstallHome` es el directorio raíz de la instalación especificado al instalar el producto (de manera predeterminada, es `home-directory/MessageQueue`).

La estructura de directorios de Message Queue resultante es la misma que en Windows (consulte la sección Windows del [Apéndice A, “Distribution-Specific Locations of Message Queue Data” de Sun GlassFish Message Queue 4.4 Administration Guide.](#))

Message Queue es compatible con la plataforma AIX, incluida la C-API de Message Queue. Si necesita instrucciones para generar y compilar aplicaciones C en la plataforma AIX, consulte XREF.

## Nuevo instalador basado en Zip

Message Queue 4.3 incluye un nuevo instalador para distribuciones basadas en zip, en lugar de distribuciones de paquete nativo. El instalador permite instalar las nuevas distribuciones .zip de Message Queue para la plataforma AIX.

El nuevo instalador extrae los archivos .zip de Message Queue en cualquier directorio para el que el usuario tenga derechos de escritura (no son necesarios privilegios root) y también permite registrar la instalación de Message Queue con Sun Connection.

Para minimizar el tamaño de los paquetes de descarga, Java Runtime no se incluye en la distribución basada en zip (la mayoría de sitios ya lo tienen). Por lo tanto, el comando `installer` requiere que se especifique un JDK o JRE, ya sea mediante la variable de entorno `JAVA_HOME` o utilizando la opción `-j` en la línea de comando, como se muestra a continuación:

```
$ installer -j ruta-JDK/JRE
```

`ruta-JDK/JRE` es la ruta del JDK o JRE especificado.

## Compatibilidad con plataformas ampliada

Message Queue 4.3 será compatible con las siguientes plataformas adicionales:

- Oracle 11g
- Windows Server 2008

## Mejoras adicionales

Message Queue 4.3 incluye las siguientes mejoras adicionales:

- “Nueva estructura de directorios en Windows” en la página 18
- “Nuevas propiedades de agente” en la página 18
- “Mejoras de la API de administración JMX” en la página 18
- “Listado de suscripciones duraderas de suscriptores comodín” en la página 19

## Nueva estructura de directorios en Windows

La estructura de directorios de la instalación de Message Queue en Windows se ha modificado con respecto a versiones anteriores, para que sea igual que en la plataforma AIX. En el futuro, esta estructura de directorios se utilizará también en Solaris y Linux para facilitar el uso de varias instalaciones en un único equipo y las actualizaciones automáticas de Message Queue mediante Sun Connection; un servicio de Sun que facilita el control, la organización y el mantenimiento de hardware y software de Sun (consulte “[Compatibilidad del instalador con el registro en Sun Connection](#)” en la página 21).

## Nuevas propiedades de agente

Pueden utilizarse las siguientes propiedades nuevas para configurar un agente:

TABLA 1-6 Propiedades de envío y enrutamiento de agente

Propiedad	Tipo	Valor predeterminado	Descripción
<code>imq.transaction.producer.maxNumMsgs</code>	Entero	1000	El número máximo de mensajes que un productor puede procesar en una única transacción. Se recomienda que el valor sea inferior a 5000 para evitar que se agoten los recursos.
<code>imq.transaction.consumer.maxNumMsgs</code>	Entero	100	El número máximo de mensajes que un consumidor puede procesar en una única transacción. Se recomienda que el valor sea inferior a 1000 para evitar que se agoten los recursos.
<code>imq.persist.jdbc.connection.limit</code>	Entero	5	El número máximo de conexiones que pueden estar abiertas en la base de datos.

## Mejoras de la API de administración JMX

Se han añadido claves de datos compuestas y un atributo nuevo a la API JMX:

- Se ha añadido un atributo `NextMessageID` al `Destination Monitor MBean` para proporcionar el ID de mensaje JMS del siguiente mensaje que debe enviarse a un consumidor.
- Se ha añadido una clave `NextMessageID` para datos compuestos al `Consumer Manager Monitor MBean` para proporcionar el ID de mensaje JMS del siguiente mensaje que debe enviarse al consumidor.
- Se ha añadido una clave `NumMsgsPending` para datos compuestos al `Consumer Manager Monitor MBean` para proporcionar el número de mensajes que se han enviado al consumidor.

Si necesita más información, consulte el [Capítulo 3, “Message Queue MBean Reference”](#) de *Sun GlassFish Message Queue 4.4 Developer’s Guide for JMX Clients*.

## Listado de suscripciones duraderas de suscriptores comodín

El comando para obtener una lista de las suscripciones duraderas:

```
list dur [-d topicName]
```

se ha mejorado para poder especificar el nombre de tema. Si no se especifica ningún tema, el comando muestra todas las suscripciones duraderas de todos los temas, incluidas las que contengan caracteres comodín.

## Nuevas funciones de Message Queue 4.2

Message Queue 4.2 es una versión secundaria que incluye varias funciones nuevas, funciones mejoradas y errores corregidos. En esta sección se describen las nuevas funciones de la versión 4.2 y se ofrecen más referencias para su uso:

- “Varios destinos para un editor o suscriptor” en la página 19
- “Validación del esquema de mensajes de carga de datos XML” en la página 20
- “Admisión de C-API para transacciones distribuidas ” en la página 20
- “Compatibilidad del instalador con el registro en Sun Connection” en la página 21
- “Admisión de bases de datos MySQL” en la página 21
- “Mejoras adicionales” en la página 21

Para obtener información sobre las funciones que se incluyen en Message Queue 4.1 y 4.0, consulte “Nuevas funciones de Message Queue 4.1” en la página 22 y “Nuevas funciones de Message Queue 4.0” en la página 26, respectivamente.

## Varios destinos para un editor o suscriptor

Con Message Queue 4.2, un editor puede publicar mensajes en varios destinos de temas y un suscriptor puede consumir mensajes de varios destinos de temas. Esta función se logra al utilizar un nombre de destino de temas que incluye caracteres comodín que representen varios destinos. Al utilizar dichos nombres simbólicos, los administradores pueden crear destinos de temas adicionales, según sea necesario, coherentes con el esquema de asignación de nombres comodín. Los editores y suscriptores publican y consumen automáticamente a partir de los destinos añadidos. (Los suscriptores de los temas comodín son más comunes que los editores.)

---

**Nota** – Esta función no se aplica a destinos de cola.

---

Puede consultar el formato de los nombres de destinos de temas simbólicos y ver ejemplos de uso en la sección “Supported Topic Destination Names” de *Sun GlassFish Message Queue 4.4 Administration Guide*.

## Validación del esquema de mensajes de carga de datos XML

Esta nueva función de Message Queue 4.2, permite validar el contenido de un mensaje XML de texto (no de objeto) con un esquema XML, cuando el mensaje se envía al agente. La ubicación del esquema XML (XSD) se especifica como propiedad de un destino de Message Queue. Si no se especifica ninguna ubicación XSD, la declaración DTD del documento XML se utilizará para realizar la validación DTD. (la validación XSD, que incluye la validación del intervalo de valores y el tipo de datos, es más rigurosa que la validación DTD.)

Para obtener información sobre el uso de esta función, consulte [“Schema Validation of XML Payload Messages” de Sun GlassFish Message Queue 4.4 Developer’s Guide for Java Clients.](#)

## Admisión de C-API para transacciones distribuidas

Según el modelo de transacciones distribuidas X/Open, la compatibilidad con transacciones distribuidas se basa en un administrador de transacciones de este tipo que supervisa y administra las operaciones que realizan uno o más administradores de recursos. Con Message Queue 4.2 la C-API de Message Queue es compatible con la interfaz XA (entre un administrador de transacciones distribuidas y Message Queue como administrador de recursos compatibles con XA), lo que permite a los clientes de la C-API de Message Queue funcionar en un entorno de procesamiento de transacciones distribuidas (tales como BEA Tuxedo) con el fin de participar en transacciones distribuidas.

Esta compatibilidad con transacciones distribuidas cuenta con las siguientes nuevas funciones de C-API (y nuevos parámetros y códigos de error) que se utilizan para implementar la especificación de la interfaz XA:

```
MQGetXAConnection()  
MQCreateXASession()
```

Si la aplicación de un cliente C se va a utilizar en el contexto de una transacción distribuida, debe obtener una conexión a través de `MQGetXAConnection()` y crear una sesión para producir y consumir mensajes mediante `MQCreateXASession()`. El inicio, la confirmación y la anulación de cualquier transacción distribuida se administra a través de interfaces API suministradas por el administrador de transacciones distribuidas.

Si necesita más información sobre el uso de funciones de transacción distribuida, consulte la sección [“Working With Distributed Transactions” de Sun GlassFish Message Queue 4.4 Developer’s Guide for C Clients.](#)

Message Queue 4.2 incluye ejemplos de programación basados en el administrador de transacciones Tuxedo. Si necesita información sobre el uso de estos programas de ejemplo, consulte la sección [“Distributed Transaction Sample Programs” de Sun GlassFish Message Queue 4.4 Developer’s Guide for C Clients.](#)

---

**Nota** – La función de transacción distribuida se admite en Solaris, Linux y Windows, aunque hasta ahora sólo se ha certificado para la plataforma Solaris.

---

## Compatibilidad del instalador con el registro en Sun Connection

El instalador de Message Queue se ha mejorado para permitir el registro de Message Queue en Sun Connection, un servicio de Sun que le ayuda a supervisar, organizar y mantener software y hardware de Sun.

Puede registrar Message Queue en Sun Connection al instalarlo. La información acerca del Message Queue instalado, como por ejemplo la versión, el nombre del sistema, el sistema operativo, la fecha de instalación y otra información básica, se transmite de forma segura a la base de datos de Sun Connection. El servicio de inventario de Sun Connection puede ayudarle a organizar su hardware y software de Sun, mientras que el servicio de actualización puede informarle de las últimas soluciones de seguridad disponibles, actualizaciones recomendadas y mejoras de funciones.

Para obtener más información sobre el registro de Message Queue con Sun Connection, consulte *Sun GlassFish Message Queue 4.3 Installation Guide*.

## Admisión de bases de datos MySQL

Message Queue 4.2 es compatible con bases de datos MySQL como almacén de datos basado en JDBC. MySQL Cluster Edition puede usarse como una base de datos JDBC para un agente independiente y como el almacén de datos de alta disponibilidad necesario para una agrupación de agentes mejorada. Si necesita información sobre cómo configurar Message Queue para utilizar MySQL, consulte la sección “[Configuring a JDBC-Based Data Store](#)” de *Sun GlassFish Message Queue 4.4 Administration Guide* y “[Enhanced Broker Cluster Properties](#)” de *Sun GlassFish Message Queue 4.4 Administration Guide*.

## Mejoras adicionales

Además de las funciones descritas anteriormente, Message Queue 4.2 incluye las siguientes mejoras:

- **Estadísticas de mensaje generadas remotamente**

Message Queue 4.4 Update 1 incluye nuevos métodos de medición de destino que pueden resultar útiles a la hora de controlar destinos en una agrupación de agentes. En una agrupación de agentes, los mensajes almacenados en un destino específico de un agente determinado de la agrupación son mensajes generados directamente para enviarlos al destino, así como mensajes enviados al destino desde agentes remotos de la agrupación. Al analizar el enrutamiento y la entrega de mensajes en una agrupación de agentes, a veces resulta útil saber cuántos mensajes de un destino son locales (se han generado localmente) y cuántos son remotos (se han generado de forma remota).

Message Queue 4.2 incluye dos nuevas cantidades de medición de destino físico: Num messages remote y Total Message bytes remote. Puede acceder a las nuevas cantidades de medición con los comandos `imqcmd list dst` e `imqcmd query dst` (consulte la sección “[Viewing Physical Destination Information](#)” de *Sun GlassFish Message Queue 4.4 Administration Guide*) o usando nuevos atributos JMX (consulte la sección “[Destination Monitor](#)” de *Sun GlassFish Message Queue 4.4 Developer’s Guide for JMX Clients*).

- **Información sobre el generador y el consumidor de comodines**

Los nuevos datos de control ofrecen información que permite el uso de caracteres comodín en nombres de destino (consulte “[Varios destinos para un editor o suscriptor](#)” en la página 19). Por ejemplo, puede ver el número de generadores y consumidores de comodines asociados con un destino con el comando `imqcmd query dst` (consulte la sección “[Viewing Physical Destination Information](#)” de *Sun GlassFish Message Queue 4.4 Administration Guide*) o utilizando los nuevos atributos JMX (consulte la sección “[Destination Monitor](#)” de *Sun GlassFish Message Queue 4.4 Developer’s Guide for JMX Clients*). La información de comodines también puede obtenerse mediante los MBeans `ConsumerManager Monitor` y `ProducerManager Monitor`.

- **Admisión del formato de nombre de usuario DN para la autenticación del cliente**

Message Queue 4.2 admite el formato de nombre de usuario DN al autenticar la conexión del cliente con un almacén de usuarios LDAP. La compatibilidad incluye la siguiente nueva propiedad de agente (y el valor):

```
imq.user_repository.ldap.usrformat=dn
```

Esta propiedad permite al agente autenticar un usuario del cliente con un almacén de usuarios LDAP extrayendo del formato de nombre de usuario DN el valor del atributo especificado por la siguiente propiedad:

```
imq.user_repository.ldap.uidattr
```

El agente utiliza el valor del anterior atributo como nombre de usuario en las operaciones de control de acceso.

Por ejemplo, si `imq.user_repository.ldap.uidattr=udi` y un nombre de usuario de autenticación de cliente están en el formato

```
udi=mquser,ou=People,dc=red,dc=sun,dc=com,
```

se extraería “mquser” para realizar el control de acceso.

- **Mejora de la autenticación JAAS**

Message Queue 4.2 permite la autenticación JAAS por dirección IP y por nombre de usuario.

## Nuevas funciones de Message Queue 4.1

Message Queue 4.1 es una versión secundaria que incluye varias funciones nuevas, funciones mejoradas y errores corregidos. Esta sección describe las nuevas funciones de la versión 4.1 y le ofrece más referencias para su uso:

- “Agrupaciones de agentes de alta disponibilidad” en la página 23
- “Compatibilidad con JAAS” en la página 24
- “Cambio del formato del almacén de datos persistente” en la página 24
- “Configuración del entorno del agente” en la página 25
- “Compatibilidad con Java ES Monitoring Framework” en la página 25
- “Administración de transacciones mejorada” en la página 25
- “Puertos fijos para las conexiones de los clientes C” en la página 25

Para obtener información sobre las funciones nuevas de Message Queue 4.0, consulte “[Nuevas funciones de Message Queue 4.0](#)” en la página 26.

## Agrupaciones de agentes de alta disponibilidad

Message Queue 4.1 incluye una agrupación de agentes nueva y mejorada. En comparación con una agrupación de agentes convencional, que sólo ofrece disponibilidad de *servicio de mensajería* (si un agente falla, hay otro agente disponible para ofrecer el servicio de mensajería), las agrupaciones de agentes de alta disponibilidad también incluyen disponibilidad de *datos* (si un agente falla, están disponibles sus mensajes persistentes y sus datos de estado para que otro agente los utilice con el fin de hacerse cargo del envío de mensajes).

El sistema de alta disponibilidad que se utiliza en Message Queue 4.1 utiliza un almacén de datos compartido basado en JDBC: en lugar de que cada agrupación de agentes tenga su propio almacén de datos persistente, todos los agentes de la agrupación comparten la misma base de datos JDBC. Si un agente en particular tiene algún error, otro agente de la agrupación asume el envío del mensaje del agente que ha fallado. Al hacer esto, el agente sustituto utiliza datos e información sobre el estado del almacén de datos compartido. Los clientes de mensajería del agente que ha fallado se reconectan al agente de sustitución, que ofrece un servicio de mensajería sin interrupciones.

El almacén compartido basado en JDBC utilizado en la implementación de alta disponibilidad de Message Queue 4.1 debe ser de por sí altamente disponible. Si no cuenta con una base de datos de alta disponibilidad o si el envío de mensajes ininterrumpido no le resulta importante, puede continuar utilizando agrupaciones convencionales, que ofrecen disponibilidad de servicios sin disponibilidad de datos.

Para configurar una agrupación de agentes mejorada en Message Queue 4.1, debe especificar las siguientes propiedades de cada agente de la agrupación:

- *Propiedades de pertenencia a la agrupación*, que sirven para especificar si el agente forma parte de una agrupación de agentes mejorada, el ID del clúster y el ID del agente del clúster.
- *Propiedades de la base de datos de alta disponibilidad*, que especifican el modelo de datos persistentes (JDBC), el nombre del proveedor de la base de datos y las propiedades de configuración específicas del proveedor.
- *Detección de fallos y propiedades de error*, que especifica cómo se detectan fallos del agente y cómo se administran mediante un agente de sustitución.

Para utilizar el sistema de agrupación de agentes mejorada, debe hacer lo siguiente:

1. Instalar una base de datos de alta disponibilidad.
2. Instalar el archivo .jar del controlador de JDBC.
3. Crear el esquema de la base de datos para el almacén de datos persistente de alta disponibilidad.
4. Establecer propiedades de alta disponibilidad para cada agente de la agrupación.
5. Iniciar cada uno de los agentes de la agrupación.

Si necesita una explicación del concepto de agrupaciones de agentes mejoradas en relación a las agrupaciones convencionales, consulte el [Capítulo 4, “Broker Clusters” de \*Sun GlassFish Message Queue 4.4 Technical Overview\*](#). Si necesita información sobre los procedimientos y datos de referencia sobre las agrupaciones de agentes mejoradas, consulte el [Capítulo 10, “Configuring and Managing Broker Clusters” de \*Sun GlassFish Message Queue 4.4 Administration Guide\*](#) y la sección “Cluster Configuration Properties” de [Sun GlassFish Message Queue 4.4 Administration Guide](#).

Si ha estado utilizando una base de datos de alta disponibilidad con Message Queue 4.0 y desea cambiar a una agrupación de agentes mejorada, puede utilizar la herramienta Database Manager (`imqdbmgr`) para convertir el sistema en un almacén de datos persistente compartido. Consulte también [“Agrupaciones de agentes” en la página 52](#) si necesita más información sobre problemas conocidos y limitaciones.

## Compatibilidad con JAAS

Además de los mecanismos de autenticación basados en archivos y en LDAP integrados, Message Queue 4.1 también es compatible con JAAS (Java Authentication and Authorization Service), que le permite conectar un mecanismo de autenticación externo al agente para autenticar los clientes de Message Queue.

Para obtener una descripción de la información que un agente pone a su disposición en un servicio de autenticación compatible con JAAS y una explicación acerca de cómo configurar el agente para utilizar dicho servicio, consulte [“Using JAAS-Based Authentication” de \*Sun GlassFish Message Queue 4.4 Administration Guide\*](#).

## Cambio del formato del almacén de datos persistente

El almacén de datos basado en JDBC de Message Queue 4.1 se cambió para admitir agrupaciones de agentes mejoradas. Por este motivo, el formato del almacén de datos basado en JDBC se ha ampliado a la versión 410. Las versiones con formato 350, 370 y 400 se migran automáticamente a la versión 410.

Tenga en cuenta que el formato del almacén de datos persistente basado en archivos permanece en la versión 370 dado que no se realizaron cambios en el mismo.

## Configuración del entorno del agente

La propiedad `IMQ_DEFAULT_EXT_JARS` se ha añadido al archivo de configuración del entorno de Message Queue 4.1, `imqenv.conf`. Puede configurar esta propiedad para especificar el nombre de las rutas de los archivos `.jar` que deben incluirse en `CLASSPATH` al iniciarse el agente. Si utiliza esta propiedad para especificar la ubicación de los archivos `.jar` externos, ya no necesitará copiar estos archivos en el directorio `lib/ext`. Los archivos `.jar` externos pueden hacer referencia a los controladores de JDBC o a módulos de registro JAAS. La siguiente propiedad de ejemplo especifica la ubicación de los controladores JDBC.

```
IMQ_DEFAULT_EXT_JARS=/opt/SUNWhadb4/lib/hadbjdbc4.jar:/opt/SUNWjavadb/derby.jar
```

## Compatibilidad con Java ES Monitoring Framework

Message Queue 4.1 es compatible con Sun Java Enterprise System (Java ES) Monitoring Framework, que permite controlar los componentes de Java ES mediante una interfaz gráfica común. La interfaz se implementa mediante una consola basada en web llamada "Sun Java System Monitoring Console". Los administradores pueden utilizar la Consola para ver las estadísticas de rendimiento, crear reglas para realizar un control automático y alarmas de confirmación. Si ejecuta Message Queue junto con otros componentes de Java ES, le resultará más cómodo gestionar todos los componentes con una única interfaz.

Para obtener más información sobre cómo utilizar la función Java ES Monitoring Framework para controlar Message Queue, consulte XREF.

## Administración de transacciones mejorada

Anteriormente, los administradores sólo podían deshacer transacciones que estuvieran en estado `PREPARED`. Es decir, si una sesión que formara parte de una transacción distribuida no terminaba correctamente, la transacción permanecía en un estado que era imposible de reorganizar para el administrador. Con Message Queue 4.1, el usuario puede utilizar la herramienta Command (`imqcmd`) para limpiar (deshacer) transacciones que se encuentren en los siguientes estados: `STARTED`, `FAILED`, `INCOMPLETE`, `COMPLETE` y `PREPARED`.

Para poder determinar si una transacción puede deshacerse (especialmente si no se encuentra en un estado `PREPARED`), la herramienta Command proporciona datos adicionales en la información `imqcmd query txn` generada: el id de la conexión que ha iniciado la transacción y la hora a la que se creó la misma. Basándose en esta información, el administrador puede decidir si es preciso deshacer o no la transacción. Por lo general, el administrador debe evitar deshacer una transacción prematuramente.

## Puertos fijos para las conexiones de los clientes C

En Message Queue 4.1, los clientes C, como los clientes Java, pueden conectarse a un puerto de agente fijo en lugar de a un puerto asignado dinámicamente por el servicio Port Mapper del agente. Las conexiones a puertos fijos son útiles si intenta utilizar un servidor de seguridad o si necesita omitir el servicio Port Mapper por alguna otra razón.

Para configurar una conexión a puerto fija, debe configurar el agente y el tiempo de ejecución del cliente C (ambos extremos de la conexión). Por ejemplo, si desea conectar a su cliente con el puerto 1756 mediante `ssljms`, tendría que hacer lo siguiente:

- En el cliente, establezca las siguientes propiedades:  
`MQ_SERVICE_PORT_PROPERTY=1756`  
`MQ_CONNECTION_TYPE_PROPERTY=SSL`
- En el agente, establezca la propiedad `imq.serviceName.protocolType.puerto` de la siguiente manera:  
`imq.ssljms.tls.port=1756`

---

**Nota** – La propiedad de conexión `MQ_SERVICE_PORT_PROPERTY` ha pasado al puerto anterior de Message Queue 3.7 Update 2.

---

## Nuevas funciones de Message Queue 4.0

Message Queue 4.0 es una versión secundaria que se limita a ser compatible con Application Server 9 PE. Incluía algunas nuevas funciones, algunas mejoras relacionadas con las funciones y soluciones a fallos. Esta sección incluye una descripción de las nuevas funciones de esta versión:

- [“Compatibilidad con la API de administración JMX” en la página 26](#)
- [“Registro del tiempo de ejecución del cliente” en la página 27](#)
- [“API de notificación de sucesos de conexión” en la página 27](#)
- [“Mejoras en la administración del agente” en la página 27](#)
- [“Mostrar información sobre el almacén de datos basado en JDBC” en la página 28](#)
- [“Compatibilidad con el proveedor de JDBC” en la página 29](#)
- [“Cambios en el formato del almacén de datos persistente.” en la página 29](#)
- [“Propiedades de mensajes adicionales” en la página 29](#)
- [“Compatibilidad con SSL” en la página 29](#)



---

**Precaución** – Uno de los cambios más insignificantes que incorporaba la versión 4.0, pero que puede provocar interrupciones, era la anulación de la opción de línea de comandos para especificar una contraseña. De ahora en adelante, debe guardar todas las contraseñas en un archivo tal y como se describe en [“Opción de contraseña eliminada” en la página 48](#), o introducirlas cuando se le soliciten.

---

## Compatibilidad con la API de administración JMX

Se ha añadido una nueva API a Message Queue 4.0 para configurar y supervisar los agentes de Message Queue de acuerdo con la especificación Java Management Extensions (JMX). Con esta API, puede configurar y supervisar las funciones de agente de forma práctica desde una

aplicación Java. En versiones anteriores de Message Queue, sólo era posible acceder a estas funciones desde las utilidades de administración de línea de comando o la consola de administración.

Si necesita más información, consulte la [Sun GlassFish Message Queue 4.4 Developer's Guide for JMX Clients](#).

## Registro del tiempo de ejecución del cliente

Message Queue 4.0 incluía compatibilidad con el registro de conexión del tiempo de ejecución del cliente y eventos relacionados con las sesiones.

Para obtener más información sobre el registro del tiempo de ejecución del cliente y cómo configurarlo, consulte la página 137 de la Guía para desarrolladores de Java.

## API de notificación de sucesos de conexión

Message Queue 4.0 incluía una API de notificación de sucesos que permitía que el tiempo de ejecución de cliente informase a una aplicación sobre los cambios en el estado de la conexión. Las notificaciones de los sucesos de conexión permiten al cliente de Message Queue escuchar los sucesos de cierre y reconexión, y tomar las medidas oportunas en función del tipo de notificación y del estado de conexión. Por ejemplo, si se produce un error y el cliente se vuelve a conectar a otro agente, es posible que una aplicación quiera limpiar el estado de su transacción y continuar con una nueva.

Para obtener información acerca de los eventos de conexión y sobre cómo crear un receptor de eventos, consulte la página 96 de la Guía para desarrolladores de Java.

## Mejoras en la administración del agente

En Message Queue 4.0, se añadieron varias opciones de comando y un nuevo subcomando a la utilidad Command (`imqcmd`) para permitir a los administradores desactivar un agente, cerrar un agente tras un intervalo específico, destruir una conexión o establecer propiedades del sistema java (por ejemplo, propiedades relacionadas con la conexión).

- Al desactivar el agente, éste pasa a un estado inactivo, lo cual permite drenar los mensajes antes de cerrar o reiniciar el agente. No es posible crear conexiones con un agente que está desactivado. Para desactivar el agente, introduzca un comando como el siguiente:

```
imqcmd quiesce bkr -b Wolfgang:1756
```

- Para que el agente se cierre después del intervalo especificado, introduzca un comando como el siguiente: (El intervalo de tiempo especifica el número de segundos que debe transcurrir para que se cierre el agente.)

```
imqcmd shutdown bkr -b Hastings:1066 -time 90
```

Si especifica un intervalo de tiempo, el agente registrará un mensaje indicando cuándo se producirá el cierre. Por ejemplo:

El agente se cerrará en 29 segundos (29996 milisegundos)

Mientras el agente espera a que se produzca el cierre, su comportamiento se verá afectado de la siguiente forma:

- Seguirá aceptando las conexiones jms administrativas.
- No aceptará nuevas conexiones jms.
- Las conexiones jms existentes continuarán funcionando.
- El agente no podrá sustituir a ningún otro agente de una agrupación de agentes mejorada.
- La utilidad `imqcmd` no se bloqueará, sino que enviará la solicitud de cerrar el agente e informará inmediatamente.
- Para destruir una conexión, introduzca un comando como el siguiente:

```
imqcmd destroy cxn -n 2691475382197166336
```

Utilice los comandos `imqcmd list cxn` o `imqcmd query cxn` para obtener el ID de conexión.

- Para establecer una propiedad con `imqcmd`, utilice la nueva opción `-D`. Es útil para establecer o anular las propiedades de fábrica de la conexión JMS o las propiedades del sistema java relacionadas con la conexión. Por ejemplo:

```
imqcmd list svc -secure -DimqSSLIsHostTrusted=true
imqcmd list svc -secure -Djavax.net.ssl.trustStore=/tmp/mytruststore
-Djavax.net.ssl.trustStorePassword=mytrustword
```

Encontrará toda la información sobre la sintaxis del comando `imqcmd` en el [Capítulo 16, “Command Line Reference”](#) de *Sun GlassFish Message Queue 4.4 Administration Guide*.

## Mostrar información sobre el almacén de datos basado en JDBC

En Message Queue 4.0 se añadió un nuevo subcomando `query` a la utilidad Database Manager, `imqdbmgr`. Este subcomando se utiliza para mostrar información sobre el almacén de datos basado en JDBC, como la versión de la base de datos, el usuario de la base de datos y si se han creado las tablas de la base de datos.

A continuación, se muestra un ejemplo de la información mostrada por este comando.

```
imqdbmgr query
```

```
[04/Oct/2005:15:30:20 PDT] Using plugged-in persistent store:
    version=400
    brokerid=Mozart1756
    database connection url=jdbc:oracle:thin:@Xhome:1521:mqdb
    database user=scott
Running in standalone mode.
Database tables have already been created.
```

## Compatibilidad con el proveedor de JDBC

En Message Queue 4.0, Apache Derby Version 10.1.1 ya es compatible como proveedor de almacén de datos basado en JDBC.

## Cambios en el formato del almacén de datos persistente.

Message Queue 4.0 introdujo cambios en el almacén de datos basado en JDBC para optimizar el sistema y permitir mejoras en el futuro. Por este motivo, el formato del almacén de datos basado en JDBC se amplió a la versión 400. Tenga en cuenta que en Message Queue 4.0, la versión del almacén de datos basado en archivos sigue siendo la 370, porque no se ha realizado ningún cambio.

## Propiedades de mensajes adicionales

Message Queue 4.0 añadió dos nuevas propiedades que están configuradas en todos los mensajes ubicados en la cola de mensajes inactivos.

- `JMS_SUN_DMQ_PRODUCING_BROKER` indica al agente dónde se generó el mensaje.
- `JMS_SUN_DMQ_DEAD_BROKER` indica al agente quién marcó el mensaje como inactivo.

## Compatibilidad con SSL

Desde la versión Message Queue 4.0, el valor predeterminado de la propiedad de fábrica de la conexión del cliente `imqSSLIsHostTrusted` es `false`. Si su aplicación depende del valor predeterminado anterior `true`, deberá cambiar la configuración y establecer la propiedad explícitamente en `true`.

Puede elegir si desea confiar en el host cuando el agente está configurado para utilizar certificados autofirmados. En este caso, además de especificar que la conexión debe usar un servicio de conexión basado en SSL (con la propiedad `imqConnectionType`), debería establecer la propiedad `imqSSLIsHostTrusted` en `true`.

Por ejemplo, para ejecutar de forma segura las aplicaciones del cliente cuando el agente utiliza certificados autofirmados, utilice un comando como el siguiente:

```
java -DimqConnectionType=TLS
      -DimqSSLIsHostTrusted=true ClientAppName
```

Para utilizar la utilidad Comando (`imqcmd`) de forma segura cuando el agente utiliza certificados autofirmados, utilice un comando como el siguiente (para enumerar los servicios del conector).

```
imqcmd list svc -secure -DimqSSLIsHostTrusted=true
```

## Funciones que se eliminarán en futuras versiones

Las siguientes funciones nuevas se dejarán de utilizar en una versión futura:

- **Supervisión basada en mensajes**

La supervisión basada en mensajes utiliza el productor de mensajes de medición (Metrics Message Producer) configurable del agente para escribir datos de medición en mensajes JMS, que se envían a continuación a destinos de temas de medición, en función del tipo de información sobre medición que contengan los mensajes. Puede acceder a esta información sobre medición escribiendo una aplicación de cliente que se suscriba al destino de temas de medición adecuado, consuma sus mensajes y procese los datos según desee.

La función de supervisión basada en mensajes ha sido sustituida por la API de administración JMX que se utilizaba en MQ 4.0 (consulte [“Compatibilidad con la API de administración JMX” en la página 26](#)). La API JMX es más completa (incluye más datos sobre medición que los que aparecen en los destinos de temas) y está basada en el estándar del sector JMX.

No hay razón de peso para utilizar la supervisión basada en mensajes ahora que Message Queue es compatible con la API JMX. La información sobre la supervisión basada en mensajes permanecerá en la documentación de Message Queue hasta que la función se cancele formalmente.

- **Programa de instalación basado en texto**

El modo de texto del instalador de Message Queue (`installer -t`) se eliminará en todos los sistemas operativos. En el modo de texto, se muestra texto sin formato en la ventana de terminal para simular el aspecto del modo de interfaz gráfica de usuario (GUI). El modo de interfaz gráfica y el modo silencioso seguirán activos.

- **Compatibilidad con plataformas**

Windows 2000 y Red Hat Linux 3 ya no se admitirán en futuras versiones.

- **Adaptador de recursos JMSRA**

El adaptador de recursos de Message Queue `imqjmsra.rar`, normalmente llamado JMSRA, se reemplazará por un nuevo adaptador de recursos en la próxima versión de Message Queue. JMSRA se utiliza para integrar Message Queue con el servidor de aplicaciones Sun Java System Application Server.

El nuevo adaptador de recursos, que combinará las funciones de JMSRA con las de otros adaptadores Sun JMS, será específicamente compatible con Message Queue, así como con otros proveedores, en un entorno de servidor de aplicaciones Java EE 5. Como tal, se utilizará para integrar Message Queue en Sun GlassFish Enterprise Server y Sun Java Composite Application Platform Suite (Java CAPS).

# Errores solucionados en Message Queue 4.4 Update 1 y versiones recientes

Message Queue 4.4 Update 1 incluye nuevas correcciones de errores, así como correcciones de errores de versiones anteriores de la familia Message Queue 4.x.

Las siguientes secciones enumeran errores que se resolvieron en sus respectivas versiones:

- “Errores solucionados en Message Queue 4.4 Update 1” en la página 31
- “Errores solucionados en Message Queue 4.4” en la página 33
- “Errores solucionados en Message Queue 4.3” en la página 36
- “Errores solucionados en Message Queue 4.2” en la página 36
- “Errores solucionados en Message Queue 4.1” en la página 38
- “Errores solucionados en Message Queue 4.0” en la página 39

## Errores solucionados en Message Queue 4.4 Update 1

En la siguiente tabla se muestran los errores solucionados en Message Queue. Algunos de estos problemas están marcados con "(OpenMQ)", que indica el problema se ha arreglado en el proyecto de código abierto [Open Message Queue](#) en el que está basado Sun GlassFish Message Queue.

TABLA 1-7 Errores solucionados en Message Queue 4.4 Update 1

Error	Descripción
6590909	El modo DIRECT MDB no se conecta al agente remoto si se anula addresslist
6616704	Aumenta el tamaño de la memoria del agente cuando se crean muchos consumidores en una sesión
6745761	XAResource.isSameRM() debe devolver el valor true cuando se usan dos conexiones en el mismo XA TX (con JMSJCA)
6745763	XAResource.isSameRM() debe devolver el valor true cuando se usan dos conexiones en el mismo XA TX (modo JMSRA DIRECT)
6745768	XAResource.isSameRM() debe devolver el valor true cuando se usan dos conexiones en el mismo XA TX (JMSRA LOCAL/REMOTE)
6760450	El almacén de mensajes se daña si se reinicia el equipo sin detener la instancia de MQ (GF)
6766241	UMS: el ejemplo SendMsg.html AJAX usa /ums como raíz de contexto predeterminada. Debería usar /imqums
6766852	DirectXAResource traduce el estado CONFLICT del agente a "TxID is already in use"
6799428	Mensajes no persistentes/no duraderos depositados en DMQ no pueden consumirse pero sí examinarse.

**TABLA 1-7** Errores solucionados en Message Queue 4.4 Update 1 (Continuación)

Error	Descripción
6799428	Mensajes no persistentes/no duraderos depositados en DMQ no pueden consumirse pero sí examinarse.
6809353	openmq 4.3 HA con postgresql (8.1) no funciona (imqbrokerd no puede iniciarse)
6809750	La agrupación de conexión (de JMSRA) para la conexión de clientId no funciona.
6812198	Classcast genera una excepción al supervisar utilizando estadísticas de tema MQ
6832000	La conexión JDBC MQ reapExcessConnection provoca un bloqueo por sobrecarga de la CPU
6833109	La aplicación de ejemplo JMX MQClusterMonitor genera una excepción en AIX con JDK6
6835420	Valor predeterminado de NoGCDefault calculado incorrectamente. Puede provocar un GC excesivo cuando queda poca memoria.
6852018	El mensaje de error "No se puede añadir el consumidor duradero {0}. No se definió ClientID en la conexión." es engañoso
6856991	Excepción NullPointerException después de que se reinicie el agente y después se restaura una transacción PREPARED de consumidor duradero
6874125	ADVERTENCIA: MQJMSRA_DC2001: connectionId=555670328604044289:_destroy():llamado en una conexión...
6878945	RFE: JMSBridge: permitir la especificación de un nombre de usuario/contraseña para crear una conexión de connection-factory
6881493	Los destinos temporales Admin del agente HA no deben almacenarse
6881753	RFE JMSBridge: permitir que se etiquete cada mensaje con el nombre jmsbridge antes de transferencia a destino
6884673	4,4 MQ Broker no pudo establecer conexión de agrupación con agente MQ 3.7/3.6
6886390	Los mensajes publicados de Persist/Txn enviados a DMQ pueden provocar un error de mq.sys.dmq no encontrado cuando se consumen desde DMQ
6886515	Excepción AccessControlException cuando se utiliza JMX para eliminar un destino de un agente incrustado
6890628	Definir la propiedad de agente "imq.autocreate.destination.isLocalOnly=true" no tiene ningún efecto
6891615	El selector no siempre funciona al ejecutar agente 4.3 en Glassfish
6891624	Msgs 'Remote' puede llegar a ser más grande que 'Count' en 'imqcmd list dst'
6891629	Es necesario un mensaje comprensible cuando se produce una excepción aritmética en el selector

TABLA 1-7 Errores solucionados en Message Queue 4.4 Update 1 (Continuación)

Error	Descripción
6891717	No hay una causa clara para el ack ifmq.transaction.autorollback=true,autorollback PREPARED TransactionAckExistEx
6891802	"[B4061]:No se puede usar el ID de transacción.Ya está en uso" al reiniciar agente después de adquirir tx ack remoto
6892512	Pérdida de memoria: Los destinos temporales no se eliminan de la conexión al llamar a tempDest.delete()
6895040	si masterbroker tiene temp dest,slave el agente no puede obtener uidprefix al iniciarse después de que se agote el tiempo de espera de bloqueo de uidprefix lock
6896230	El nuevo consumidor creado en masterb mientras masterb se reinicia después de sincronización con subordinados puede no propagarse a todos
6896764	El método equals de TransactionAcknowledgement es incorrecto.
6898355	Bloqueo de adquisición restablecido en init de gestor de agrupaciones al reiniciarse el agente sin esperar a que se complete la adquisición
6901405	RFE: registrar información de proveedor JDBC y propiedades si se especifican
16 (OpenMQ)	El selector no siempre funciona al ejecutar agente 4.3 en Glassfish
17 (OpenMQ)	openmq 4.3 HA con postgresql (8.1) no funciona (imqbrokerd no puede iniciarse)
22 (OpenMQ)	El instalador se refiere a un archivo binario no existente y falla
25 (OpenMQ)	Pérdida de memoria al crear TemporaryTopic.
29 (OpenMQ)	Aislamiento de intermediario
30 (OpenMQ)	Msgs 'Remote' puede llegar a ser más grande que 'Count' en 'imqcmd list dst'
31 (OpenMQ)	Se necesita un mensaje comprensible cuando se produce una excepción aritmética en el selector
32 (OpenMQ)	Solución para int-> long causa desbordamiento
33 (OpenMQ)	Instalador OpenMQ: Error "Descriptor SwiXML no válido" cuando se ejecuta en un sistema con configuración regional ja

## Errores solucionados en Message Queue 4.4

La siguiente tabla muestra los errores solucionados en Message Queue 4.4.

TABLA 1-8 Errores solucionados en Message Queue 4.4

Error	Descripción
6242247	Agrupación MQ con inicio de agente maestro (masterbroker) y se bloquea si ambos agentes se encuentran en el mismo sistema y tienen el mismo nombre
6760937	El agente no vuelve a conectarse con la base de datos si se reinicia
6763252	El agente debe registrar un mensaje más comprensible que NPE al ack un mensaje que ha caducado o se ha eliminado
6765410	Cuando el agente maestro (masterbroker) envía intereses locales dos veces provoca una excepción de subordinado "Durable subscription already active" (la suscripción duradera ya está activa)
6796506	El mensaje remoto PREPARED no se envía tras deshacer si se agota el tiempo de espera de la respuesta PREPARE remota
6807708	TemporaryDestination.delete falla si el agente maestro no se está ejecutando
6812037	RFE: pass MQ_CALLBACK_RUNTIME_ERROR to afterMessageDelivery if MQMessageListenerFunc devuelve un error
6812755	El mensaje de registro de nivel BUENO debería ser ADVERTENCIA si las devoluciones de llamada before/afterMessageDelivery devuelven un error
6816023	La excepción Message.setStringProperty() no muestra el nombre de la propiedad en la excepción de carácter no permitido
6819095	RFE: la agrupación debería permitir el establecimiento del tamaño del búfer de secuencia de entrada/salida yTcpNoDelay
6820585	'imqcmd list txn' no muestra transacciones de agrupación COMMITTED en espera de finalización por agente remoto
6820588	Una transacción de agrupación que consume mensajes locales y remotos permanece como COMMITTED en estado de espera
6821639	NPE de rollback/commitTransaction durante recuperación AS para modo MQRA-DIRECT
6823364	RFE: actualización del compilador C-API a Sun Studio 12 en Solaris
6829113	Excepción ConcurrentModificationException cuando el restablecimiento de Tuxedo TM agotó el tiempo de espera para la transacción bajo sobrecarga
6832197	ack remoto no transaccionado no debería esperar respuesta remota si el cliente no solicita ackack
6834735	Mensaje de registro confuso "Error interno de agente inesperado" cuando Tuxedo TM agota el tiempo de espera de un txn en estado START
6836364	El suscriptor de comodín no recibe el mensaje remoto si el tema se crea antes que el suscriptor

TABLA 1-8 Errores solucionados en Message Queue 4.4 (Continuación)

Error	Descripción
6836691	HA(JCAPS):mensaje ya eliminado excepción al recibir tras restauración de XA de receptor después confirmar un mensaje
6836749	HA(JCAPS):ack existe en excepción de almacén de receptor después de 1 de deshacer duraderos y después confirma un mensaje
6837671	HA(JCAPS):reenvío constante de mensaje confirmado cuando XAResourceImpl.rollback después de confirmación correcta
6839193	RFE: actualización de compilador C++ a Visual Studio 2008 SP1
6845625	Agente entra en estado de poca memoria cuando consumidores remotos se crean/cierran repetidamente
6852207	NPE al enviar mensaje a agente remoto provoca el error "imposible procesar mensaje" en el agente remoto al leer pkt de mensaje
6853822	Mensaje de excepción confuso "Imposible realizar operación END_TRANSACTION" al finalizar un txn fallido
6854142	"Esperando conexión de agrupación" "Conexión de agrupación cerrada" a agente remoto cada 3 minutos
6858121	WARNING confuso "Transacción desconocida" en registro de agente en 'imqcmd list txn' si existe txn remoto
6858488	txn COMMITTED no eliminado de agente de inicio txn si el agente del participante remoto ha eliminado su COMMITTED tx
6858905	ConcurrentModificationException en Consumer.destroyConsumer
6861362	RFE: JMSBridge: admite asignación automática de destino a origen Message.getJMSDestination
6861528	RFE: JMSBridge: permitir mensaje de ramificación MessageTransformer.transform() a un destino diferente
6861653	Información de txn de agrupación excesiva enviada a agente remoto inactivo incompleto COMMIT con sobrecarga txn
6862413	Mensaje de registro confuso "mq://xxx.xxx.xx.xx:pppp/ ..." puede alcanzarse en 60 segundos"
6863867	Excepción MissingResourceException al reiniciar agente HA si tiene COMMITTED pendiente de un agente remoto inactivo
6867596	Recuperado txn PREPARED después de que el reinicio del agente devuelva estado backto PREPARED si se vuelve a reiniciar el agente
6868525	Excepción NullPointerException al reenviar destino temporal a agente remoto en establecimiento de vínculo

**TABLA 1-8** Errores solucionados en Message Queue 4.4 (Continuación)

Error	Descripción
6868578	Algunos broadcast/unicast sin comprobación si un vínculo establecido interfiere con protocolo de enlace de vínculo porque está inactivo
6871612	HA:log msgs"Imposible notificar finalización de transacción.."al consumir mensajes remotos si el vínculo pendiente está inactivo
6886391	Excepción NullPointerException al confirmar mensaje si el mensaje ya se ha eliminado

## Errores solucionados en Message Queue 4.3

La siguiente tabla muestra los errores solucionados en Message Queue 4.3.

**TABLA 1-9** Errores solucionados en Message Queue 4.3

Error	Descripción
6634033	El protocolo de la agrupación no propaga el valor de <code>imqConsumerFlowLimit</code> a los agentes remotos cuando se crea un cliente.
6713012	La destrucción de un consumidor en un agente de una agrupación al mismo tiempo que el agente remoto se está reiniciando puede provocar que no se reciban algunos mensajes.
6727555	En el mensaje de registro de agente "Max bytes per msg exceeded" aparecen el tamaño del mensaje y el valor de max bytes per message intercambiados.
6737404	Las mediciones JMX deben proporcionar cantidades de mensajes enviados desde destinos (temas y colas) pero que todavía no han recibido los consumidores.
6740568	El agente genera una excepción cuando se consumen demasiados mensajes en una única transacción.
6758524	El comando para obtener una lista suscripciones duraderas ( <code>imqcmd lista dur -d "foo.*"</code> ) no acepta caracteres comodines en el nombre del destino.
6758952	Si se define la opción <code>imq.portmapper.hostname=localhost</code> , los agentes no pueden conectarse a una agrupación.
6758817	Si se define la opción <code>imq.cluster.hostname=localhost</code> (no recomendado) los agentes de diferentes equipos no pueden conectarse a una agrupación.

## Errores solucionados en Message Queue 4.2

La siguiente tabla muestra los errores solucionados en Message Queue 4.2.

TABLA 1–10 Errores solucionados en Message Queue 4.2

Error	Descripción
6581592	Cuando el instalador o el desinstalador se ejecutan en modo texto ( <code>installer -t</code> ), la pantalla Resumen muestra el directorio que contiene los archivos de registro y de resumen pero no enumera los nombres de estos archivos.
6585911	La pantalla Selección de JDK del instalador incluye incorrectamente el JRE que se incluye con el instalador y que se utiliza para ejecutar el instalador.
6587112	La pantalla de resumen del instalador muestra elementos no utilizados en las configuraciones regionales de varios bytes.
6587127	Al ejecutar el instalador haciendo referencia a un archivo de respuesta ( <code>installer -a nombrearchivo -s</code> ), si el archivo de respuesta no existe, los mensajes de error serán incoherentes y confusos.
6590969	Permite el formato de nombre de usuario DN en la autenticación de conexión del cliente.
6594381	La instalación de los RPM de localización de Message Queue 4.1 (que aparece si se selecciona la casilla de verificación “Instalar paquetes multilingües de Message Queue” en la pantalla Paquetes multilingües) no se realizará correctamente si ya existen en su sistema RPM de localización de Message Queue.
6599144	Cuando desinstale Message Queue 4.2, la pantalla de presentación y las pantallas y los márgenes del desinstalador aparecerán vacíos y grises en la versión Java SE 6, pero funcionarán en Java SE 5.
6615741	El mensaje entregado en una sesión de consumidor enviada que se ha deshecho no se volverá a enviar si el consumidor original se cerró antes de deshacerla.
6629922	El controlador de transacciones distribuidas no vuelve a enviar los mensajes a un consumidor inactivo en el orden correcto.
6635130	El agente no informa correctamente al productor sobre mensajes no persistentes para reanudar la producción tras haber realizado una pausa porque el destino ha alcanzado el límite de mensajes o memoria.
6641117	El mensaje enviado en una sesión de consumidor transaccionada que se ha deshecho no se volverá a entregar si el consumidor original se cerró después de deshacerla.
6683897	La pantalla de resumen del instalador de Message Queue informa sobre un error de configuración aunque parezca que se haya realizado correctamente: el instalador no puede escribir en <code>/dev/sterr</code> en algunos equipos.
6684069	En la agrupación de agentes en la que se envía un gran número de mensajes al cliente remoto en las transacciones de consumidor, la transacción de confirmación no se realiza correctamente.
6688935	El valor predeterminado del tiempo de espera de lectura de Portmapper es demasiado pequeño.

**TABLA 1-10** Errores solucionados en Message Queue 4.2 (Continuación)

Error	Descripción
6695238	Las aplicaciones de cliente C no se pueden conectar a un agente instalado en una ubicación que tenga espacios en la ruta.
6710168	El consumidor ya no consume mensajes si se realizan dos pausas en el destino sin reanudar el proceso entre las pausas.
6710169	El funcionamiento de <code>JMX ConsumerManagerMonitor.getConsumerInfo</code> siempre devuelve <code>SESSION_TRANSACTED</code> en el modo de confirmación.

## Errores solucionados en Message Queue 4.1

La siguiente tabla muestra los errores solucionados en Message Queue 4.1.

**TABLA 1-11** Errores solucionados en Message Queue 4.1

Error	Descripción
6381703	Los mensajes remotos tramitados pueden confirmarse dos veces si se reinicia el agente que lo ha originado.
6388049	No es posible deshacer una transacción distribuido incompleta
6401169	Las opciones "confirmar" y "deshacer" de <code>imqcmd</code> no muestran un mensaje de confirmación.
6473052	El valor predeterminado de las colas autocreadas debería ser "round robin" (operación por turnos). ( <code>MaxNumberConsumers = -1</code> ).
6474990	El registro del agente muestra <code>ConcurrentModificationException</code> para el comando <code>imqcmd list dst</code> .
6487413	Se produce una pérdida de memoria cuando el comportamiento límite es <code>REMOVE_OLDEST</code> o <code>REMOVE_LOWER_PRIORITY</code> .
6488340	El agente hace un "giro" y el cliente espera a que se atienda la respuesta.
6502744	El agente no cumple el límite predeterminado de 1000 mensajes de la cola de mensajes inactivos.
6517341	El tiempo de ejecución del cliente tiene que mejorar la lógica de reconexión cuando el cliente está conectado a una agrupación de agentes mejorada y permitir al cliente volver a conectarse al margen de cuál sea el valor de la propiedad <code>imqReconnectEnabled</code> .
6528736	El servicio de inicio automático de Windows ( <code>imqbrokersvc</code> ) se bloquea durante el inicio.
6561494	Los mensajes se envían al consumidor equivocado cuando ambos comparten una sesión.
6567439	Los mensajes generados en una transacción <code>PREPARED</code> se envían sin orden si se confirman después de que se reinicie el agente.

## Errores solucionados en Message Queue 4.0

La siguiente tabla muestra los errores solucionados en Message Queue 4.0.

TABLA 1-12 Errores solucionados en Message Queue 4.0

Número de error	Descripción
4986481	En Message Queue 3.5, la invocación de <code>Session . recover</code> podría bloquearse en el modo de conexión automática.
4987325	El indicador de nueva entrega se establecía en <code>false</code> para mensajes enviados de nuevo después de invocar a <code>Session . recover</code> .
6157073	Cambiado el mensaje de nueva conexión para que incluya el número de conexiones que se encuentran en el servicio además del número total de conexiones.
6193884	Message Queue genera un mensaje basura en syslog en configuraciones regionales que utilizan caracteres no ASCII en los mensajes.
6196233	No funciona la selección de mensajes con <code>JMSMessageID</code> .
6251450	<code>ConcurrentModificationException</code> en <code>connectList</code> durante cierre de agrupación.
6252763	<code>java.nio.OverflowException</code> en <code>java.nio.HeapByteBuffer .putLong/Int .</code>
6260076	El primer mensaje publicado tras el inicio es lento cuando se utiliza almacenamiento Oracle.
6260814	El procesamiento del Selector en <code>JMSXUserID</code> siempre evalúa como <code>false</code> .
6264003	El explorador de colas muestra mensajes que son parte de transacciones no confirmadas.
6271876	El control de flujo de la conexión no funciona correctamente cuando se cierra un consumidor que tiene mensajes no consumidos.
6279833	Message Queue no debería permitir que dos agentes utilicen las mismas tablas jdbc.
6293053	El agente maestro no se inicia correctamente cuando cambia la dirección IP del sistema, a menos que se borre el almacén (con <code>-reset store</code> .)
6294767	El agente de Message Queue tiene que establecer <code>SO_REUSEADDR</code> en los sockets de la red que abra.
6304949	No es posible configurar la propiedad <code>ClientID</code> de <code>TopicConnectionFactory</code> .
6307056	El registro <code>txn</code> genera un cuello de botella en el rendimiento.
6320138	La API C de Message Queue no tiene la capacidad de determinar el nombre de una cola de una cabecera de respuesta.
6320325	El agente a veces utiliza JDK 1.4 en vez de JDK 1.5 en Solaris, incluso si ambas versiones se encuentran instaladas.

TABLA 1–12 Errores solucionados en Message Queue 4.0 (Continuación)

Número de error	Descripción
6321117	Al iniciar agrupaciones de varios agentes se genera la excepción <code>java.lang.NullPointerException</code> .
6330053	El cliente de jms devuelve <code>java.lang.NoClassDefFoundError</code> al confirmar una transacción desde el suscriptor.
6340250	La API C admite el tipo MESSAGE.
6351293	Se incluye compatibilidad con la base de datos Apache Derby.

## Actualizaciones de documentación de Message Queue 4.4 Update 1

Esta sección contiene información relacionada con las actualizaciones de la documentación de Message Queue 4.4 Update 1:

- “Problemas de compatibilidad” en la página 40
- “Cambios en el conjunto de documentación de Message Queue 4.4 Update 1” en la página 41

## Problemas de compatibilidad

En esta sección se tratan los problemas de compatibilidad de Message Queue 4.4 Update 1.

### Estabilidad de interfaz

Sun GlassFish Message Queue utiliza muchas interfaces que pueden cambiar con el tiempo. El [Apéndice B, “Stability of Message Queue Interfaces” de \*Sun GlassFish Message Queue 4.4 Administration Guide\*](#) incluye una clasificación de las interfaces según su estabilidad. Cuanto más estable sea la interfaz, menos probable es que cambie en las siguientes versiones del producto.

### Problemas en relación a la próxima versión importante de Message Queue

La próxima versión importante de Message Queue puede introducir cambios que hagan que las aplicaciones de los clientes de Message Queue dejen de ser compatibles con el producto. Esta información puede ser revelada completamente.

- Las ubicaciones de los archivos instalados como parte de Sun GlassFish Message Queue pueden cambiar. Esto puede hacer que dejen de funcionar algunas de las aplicaciones que dependen de la ubicación actual de archivos de Message Queue.

- Es posible que el agente Message Queue 3.5 y agentes anteriores dejen de funcionar en agrupaciones con agentes más recientes.
- En futuras versiones, es posible que los clientes de Message Queue no puedan utilizar versiones de JDK anteriores a la 1.5.
- En futuras versiones, es posible que los clientes de Message Queue no puedan utilizar versiones de JDK anteriores a la 1.6.

## Cambios en el conjunto de documentación de Message Queue 4.4 Update 1

El conjunto de documentación de Message Queue 4.4 Update 1 incluye actualizaciones al conjunto de documentación de Message Queue 4.4 tal y como se describe a continuación:

### Descripción general técnica

La [Sun GlassFish Message Queue 4.4 Technical Overview](#) contiene las nuevas funciones de &ProductName 4.4.

### Guía de administración

La [Sun GlassFish Message Queue 4.4 Administration Guide](#) incluye pequeñas correcciones de errores y refleja nuevas funciones de Message Queue 4.4.

### Guía del desarrollador para clientes Java

La [Sun GlassFish Message Queue 4.4 Developer's Guide for Java Clients](#) incluye pequeñas correcciones de errores.

### Guía del desarrollador para clientes C

La [Sun GlassFish Message Queue 4.4 Developer's Guide for C Clients](#) incluye correcciones de errores menores.

### La Guía del desarrollador para el cliente JMX

La [Sun GlassFish Message Queue 4.4 Developer's Guide for JMX Clients](#) incluye el nombre de producto y número de versión de Message Queue.

## Limitaciones y problemas conocidos

Esta sección contiene una lista de los problemas conocidos de Message Queue 4.4 Update 1. Se describen las siguientes áreas:

- “Problemas relacionados con la instalación” en la página 42
- “Opción de contraseña eliminada” en la página 48
- “Problemas de administración y configuración” en la página 49
- “Problemas de agentes” en la página 51
- “Agrupaciones de agentes” en la página 52
- “Problemas con JMX” en la página 54
- “Compatibilidad con SOAP” en la página 55

Para consultar una lista de los fallos actuales, sus estados y sus soluciones alternativas, invitamos a los miembros de Java Developer Connection™ a visitar la página Bug Parade del sitio web Java Developer Connection. Consulte la página antes de informar sobre un nuevo error. A pesar de que no se muestran todos los problemas de Message Queue la página es un buen punto de partida para ver si un problema ha sido comunicado.

<http://bugs.sun.com/bugdatabase/index.jsp>

---

**Nota** – Aunque la suscripción a Java Developer Connection es gratuita, es necesario registrarse. Encontrará información sobre cómo ser miembro de Java Developer Connection en la página web "For Developers" de Sun.

---

Para informar de un nuevo problema o enviar una petición de nuevas funciones, envíe un mensaje a [imq-feedback@sun.com](mailto:imq-feedback@sun.com).

## Problemas relacionados con la instalación

Esta sección describe los problemas relacionados con la instalación de la versión 4.4 Update 1.

### Registro del producto y Java ES

Message Queue 4.4, al igual que Message Queue 4.2 y 4.1, se instala mediante un instalador relativamente nuevo, que a su vez instala y actualiza los componentes compartidos de Java Enterprise System (Java ES) que requiere Message Queue; como por ejemplo, JDK, NSS, JavaHelp, etc.

El nuevo instalador de Message Queue y el antiguo instalador de Java ES, que se utilizaba para instalar versiones anteriores de Message Queue, no comparten el mismo registro del producto. Si una versión de Message Queue instalada con el instalador Java ES se elimina y, más adelante, se instala Message Queue 4.4 mediante el instalador de Message Queue, el registro del producto

Java ES puede tener un estado incoherente. Como resultado, si se ejecuta el desinstalador Java ES, es posible que elimine Message Queue 4.4; y los componentes compartidos de los que depende por error, aunque dicho instalador no los haya instalado.

La mejor manera de actualizar el software Message Queue que instaló el instalador de Java ES es la siguiente.

1. Desinstale Message Queue y sus componentes compartidos con el desinstalador Java ES.
2. Vuelva a instalarlo con el instalador de Message Queue 4.4.

## Instalación en todas las plataformas

Los siguientes problemas afectan a la instalación en todas las plataformas:

- La pantalla Listo para la instalación mostrará el nombre del producto como “mq”, en lugar de como Sun Java System Message Queue 4.3. (*Fallo 6650841*)
- Cuando el instalador está en proceso de instalar Message Queue 4.3 y aparece la pantalla de progreso, el botón Cancelar está activo. Si en ese momento se pulsara el botón Cancelar, la instalación quedaría incompleta o interrumpida. (*Fallo 6595578*)
- La pantalla de resumen del instalador contiene una serie de vínculos que, al pulsar sobre ellos, activan un registro o un visor de la página de resumen. Al cerrar la ventana de este visor con el botón X de la ventana en lugar de con el botón marcado con la etiqueta "cerrar", ya no podrá volver a abrir la ventana del visor. (*Fallo 6587138*)

*Solución:* Cierre la ventana con el botón Cerrar.

- Al ejecutar el instalador en modo "sólo registro" (`installer -r`) después de realizar una instalación silenciosa en la que se ha omitido el registro, éste falla con un error "fin de archivo prematuro". (*Fallo 6767988*)
- Al ejecutar el instalador de Message Queue en un equipo sin ningún JDK instalado, aparece el siguiente mensaje de error: "Raíz no válida en clave de registro "HKLM\SOFTWARE\JavaSoft\Java Runtime Environment\CurrentVersion." (*Fallo 6764358*)

*Solución:* instale el JDK antes de ejecutar el instalador.

- El instalador de Message Queue crea el directorio mqInstallHome antes de que el usuario haga clic en el botón Listo para instalar de la pantalla Listo para instalar. (*Fallo 6595590*)

## Instalación en Windows

Al instalar Message Queue en Windows, tenga presente las siguientes limitaciones:

- La estructura de directorios de Message Queue 4.3 en la plataforma Windows es distinta a la de las versiones anteriores. Consulte “Installed Directory Structure” en la *Sun Java System Message Queue 4.3 Installation Guide*.
- El instalador no añade entradas de Message Queue en el menú Inicio>Programas. (*Fallo 6567258*)

*Solución:* Para iniciar la consola de administración, utilice la línea de comandos tal y como se indica en la sección “Starting the Administration Console” de *Sun GlassFish Message Queue 4.4 Administration Guide*.

- El instalador no añade el directorio `IMQ_HOME\mq\bin` a la variable de entorno `PATH` . (Fallo 6567197)

*Solución:* Los usuarios deben añadir esta entrada a la variable de entorno `PATH` o proporcionar un nombre de ruta completa al invocar las utilidades de Message Queue (`IMQ_HOME\mq\bin\command`).

- El instalador no agrega entradas al registro de Windows para indicar que Message Queue está instalado. (Fallo 6586389)

- El instalador no añade el agente de Message Queue como servicio de Windows.

*Solución:* añada manualmente el agente de Message Queue como servicio de Windows mediante el comando `imqsvcadm`.

- Si no hay ningún JDK instalado, el programa de instalación generará el siguiente error: "Raíz no válida en clave del registro `HKLM\SOFTWARE\JavaSoft\Java Runtime Environment\CurrentVersion`". (Fallo 6764358)\par

*Solución:* si aparece este error, instale un JDK y continúe.

- Cuando se ejecuta en modo silencio con un archivo de respuesta, el instalador rebota de inmediato. La instalación se lleva a cabo, pero el usuario no tiene forma de saber cuándo se ha completado la instalación silenciosa. (Fallo 6586560)
- El programa de instalación instala Message Queue en `C:\` incluso si el sistema operativo está instalado en una unidad diferente. (Fallo 6673511)
- Para instalar y desinstalar en Windows, no existen archivos `.bat` que el usuario pueda ejecutar ni instalar mediante la función Agregar o Quitar programas del panel de control de Windows. (Fallo 6673417)
- En Windows Vista no se puede instalar Message Queue en `C:\Archivos de programa` a menos que instale con una interfaz de línea de comandos como administrador. (Fallo 6701661)

*Solución:* Para realizar la instalación con una interfaz de línea de comandos como Administrador:

1. Inicio→Programas→Accesorios→Símbolos del sistema.
  2. Haga clic con el botón derecho del ratón en la ventana de línea de comandos.
  3. Seleccione Ejecutar como administrador.
  4. Cambie el directorio a la imagen de instalación de Message Queue 4.2.
  5. Ejecute `installer.vbs`.
- Si el desinstalador se ejecuta en modo de ejecución seca (`uninstaller -n`), realiza una desinstalación de forma incorrecta. (Fallo 6719051)

*Solución:* Realice una instalación silenciosa utilizando el siguiente comando:

```
uninstaller -s
```

- La cadena “Install Home” de la página inicial del instalador no está localizada. (*Fallo 6592491*)
- El desinstalador de Message Queue basado en zip se bloquea en 2003. (*Fallo 6764370*)  
*Solución:* eliminar manualmente el directorio *mqInstallHome*.

## Instalación en Solaris

- Si el instalador se ejecuta en modo de ejecución seca (`installer -n`), la pantalla de resumen muestra algunos mensajes de error y el estado de instalación como “incompleto”. Esto es incorrecto y confunde al usuario; una ejecución seca no instala nada en el sistema, sino que crea sólo un archivo de respuesta que puede utilizarse después para realizar la instalación silenciosa. (*Fallo 6594351*)
- El instalador no realizará el registro en Sun Connection cuando se ejecute en modo silencioso con un archivo de respuesta (`installer -a nombrearchivo -s`). (*Fallo 6710268*)

## Instalación en Linux

Estos son los problemas que se producen al hacer la instalación en la plataforma Linux:

- En Red Hat Linux 5, la biblioteca `compat-libstdc++` necesaria para ejecutar aplicaciones de cliente C no está incluida en la distribución de Message Queue y, por lo tanto, el instalador de Message Queue no la instala. Si va a desarrollar y ejecutar clientes C, deberá instalar esta biblioteca manualmente.

El rpm `compat-libstdc++` suele estar incluido en el soporte de instalación de la versión de Linux que se esté utilizando. Se puede instalar mediante el comando siguiente:

```
rpm -ivh compat-libstdc++-x.x.x.x.rpm
```

x es el número de la versión.

Para comprobar que la biblioteca se ha instalado correctamente, utilice el siguiente comando:

```
rpm -qa | grep compat-libstdc++
```

- En Red Hat Linux 5, los clientes C pueden fallar con un error `PR_LOAD_LIBRARY_ERROR` (*Error 6885978*)

En Red Hat Linux 5, los clientes C pueden fallar y mostrar un mensaje parecido a:

```
"Preparing for NSS initialization ..."
"Initializing NSS ..."
"Could not connect to broker because 'PR_LOAD_LIBRARY_ERROR' (-5977)."
```

producer(): Error: PR\_LOAD\_LIBRARY\_ERROR

Este error se produce porque las bibliotecas NSS/NSPR no están accesibles.

Para solucionar este problema, defina la variable de entorno `LD_LIBRARY_PATH` para incluir la ruta a las bibliotecas NSS/NSPR, `IMQ_HOME/nss/lib`.

- En el panel de selección de JDK, la lista desplazable sólo muestra un elemento. Esto dificulta la selección de otro JDK de la lista. (Fallo 6584735)
- Si el JDK es actual y el usuario selecciona "Instalar JDK predeterminado" en la pantalla de selección de JDK, el instalador seguirá intentando instalarlo e informará de que no puede instalar el paquete. La instalación se realiza correctamente a pesar de este problema. (Fallo 6581310)
- Si el JDK instalado actualmente es una versión anterior a JDK 1.5.0\_15 (la versión que normalmente instala el instalador de Message Queue), el desinstalador de Message Queue no podrá encontrar el directorio predeterminado `IMQ_JAVAHOME` y se producirá un error. (Fallo 6673415)

*Solución:* Instale el JDK 1.5 manualmente de la siguiente manera antes de ejecutar el desinstalador de Message Queue.

```
# cd installImage/Product/UNIX/LINUX/X86/2.4/Packages
# rpm -i --force jdk-1.5.0_15-linux- arch.rpm
arch es i586 o amd64.
```

- Si el instalador se ejecuta en modo de ejecución seca (`installer -n`), la pantalla de resumen muestra algunos mensajes de error y el estado de instalación como "incompleto". Esto es incorrecto y confunde al usuario; una ejecución seca no instala nada en el sistema, sino que crea sólo un archivo de respuesta que puede utilizarse después para realizar la instalación silenciosa. (Fallo 6594351)

## Anomalías de versión del instalador

El instalador muestra la información de la versión de Message Queue de una manera poco clara. (Fallo 6586507)

### Plataforma Solaris

En la plataforma Solaris, consulte la siguiente tabla para determinar la versión de Message Queue que aparece en el instalador.

TABLA 1-13 Traducción de la cadena de versiones

Versión que muestra el instalador en el SO Solaris	Versión de Message Queue correspondiente
4.4.1.0	4.4 Update 1
4.4.0.0	4.4
4.3.0.0	4.3
4.2.0.0	4.2

TABLA 1-13 Traducción de la cadena de versiones (Continuación)

Versión que muestra el instalador en el SO Solaris	Versión de Message Queue correspondiente
4.1.0.2	4.1 Patch 2
4.1.0.1	4.1 Patch 1
4.1.0.0	4.1
3.7.2.1	3.7 UR2 Patch 1
3.7.0.2	3.7 UR2
3.7.0.1	3.7 UR1
3.6.0.0	3.6
3.6.0.4	3.6 SP4
3.6.0.3	3.6 SP3
3.6.0.2	3.6 SP2
3.6.0.1	3.6 SP1

**Nota** – Para versiones con parche de 3.6 SP4 (por ejemplo, 3.6 SP4 Patch 1), la cadena de las versiones que muestra el instalador sigue siendo la misma. Debe ejecutar el comando `imqbrokerd -version` para determinar la versión exacta.

## Plataforma Linux

En la plataforma Linux, el número de versión que aparece en el instalador tiene el siguiente formato.

*majorReleaseNumber.minorReleaseNumber-someNumber*

Por ejemplo, 3.7-22. Esto quiere decir únicamente que se trata de una de las versiones 3.7, pero no especifica cuál. Para determinar la versión de Message Queue instalada, ejecute el comando:

`imqbrokerd -version.`

## Problemas relacionados con la traducción

Estos son los problemas relacionados con la localización.

- Cuando el instalador se ejecuta en modo texto (`installer -t`) en una configuración regional de lengua no inglesa, los caracteres de varios bytes aparecen como texto basura. (Fallo 6586923)

- En la pantalla de progreso del instalador, la barra de progreso muestra caracteres extraños. La información sobre herramientas no es modificable en configuraciones regionales de lengua no inglesa. (*Fallo 6591632*)
- La pantalla de licencia del instalador muestra el texto de licencia en inglés, al margen de cuál sea la configuración regional en la que se haya ejecutado. (*Fallo 6592399*)  
*Solución:* Para acceder a los archivos de licencia traducidos, busque el archivo `LICENSE_MULTILANGUAGE.pdf`.
- El texto de ayuda sobre el uso del instalador no está traducido. (*Fallo 6592493*)
- La cadena "Ninguno" que aparece en la página resumen del instalador aparece en inglés y no es modificable. (*Fallo 6593089*)
- Cuando el instalador se ejecuta en una configuración regional alemana, la pantalla de Bienvenida no muestra el texto completo que se ve en otras configuraciones regionales. (*Fallo 6592666*)
- La cadena "Inicio de instalación" que se ve en la página del instalador del mismo nombre no está traducida. Aparece en inglés aunque el instalador se haya ejecutado en configuraciones regionales de lengua no inglesa. (*Fallo 6592491*)
- Cuando el instalador se ejecuta en modo texto (`installer -t`), aparecen las opciones de respuesta en inglés "Yes" y "No", aunque el instalador se haya ejecutado en otra configuración regional. (*Fallo 6593230*)
- La información sobre herramientas del botón Examinar de la pantalla de selección de JDK del instalador aparece en inglés y no es modificable. (*Fallo 6593085*)

## Opción de contraseña eliminada

En versiones anteriores de Message Queue, podía usarse la opción `-p` o `-password` para especificar la contraseña de manera interactiva para los siguientes comandos: `imqcmd`, `imqbrokerd` y `imdbmgr`. A partir de la versión 4.0., estas opciones se han eliminado.

En lugar de ello, puede crear un archivo de contraseñas que especifique las contraseñas relevantes y hacer referencia al archivo de contraseñas utilizando la opción del comando `-passfile`, o simplemente introducir una contraseña cuando el comando se lo solicite.

Un archivo de contraseñas puede contener una o varias de las contraseñas enumeradas más abajo.

- Una contraseña keystore utilizada para abrir el keystore de SSL. Utilice la propiedad `imq.keystore.password` para especificar la contraseña.
- Una contraseña de depósito LDAP utilizada para conectar de forma segura con un directorio LDAP si la conexión no es anónima. Utilice la propiedad `imq.user_repository.ldap.password` para especificar esta contraseña.

- Una contraseña de base de datos JDBC utilizada para conectarse con la base de datos compatible con JDBC. Utilice la propiedad `imq.persist.jdbc.vendorName.password` para especificar esta contraseña. El componente `vendorName` del nombre de propiedad es una variable que especifica el proveedor de la base de datos. Las opciones son `hadb`, `derby`, `pointbase`, `oracle` o `mysql`.
- Una contraseña para el comando `imqcmd` (para realizar tareas administrativas del agente). Utilice la propiedad `imq.imqcmd.password` para especificar esta contraseña.

En el siguiente ejemplo, la contraseña de la base de datos JDBC se ha establecido en el archivo de contraseñas como `abracadabra`.

```
imq.persist.jdbc.mysql.password=abracadabra
```

Puede utilizar un archivo de contraseñas de cualquiera de las siguientes formas.

- Configure el agente que va a utilizar el archivo de contraseñas estableciendo las siguientes propiedades en el archivo `config.properties` del agente.
 

```
imq.passfile.enabled=true
imq.passfile.dirpath= contraseñaDirectorioArchivo
imq.passfile.name=contraseñaNombreArchivo
```
- Utilice la opción `-passfile` del comando relevante, como por ejemplo:
 

```
imqbrokerd -passfile contraseñaNombreArchivo
```

## Problemas de administración y configuración

Los siguientes problemas son generados por la administración y la configuración de Message Queue.

- En plataformas Windows, debe añadir manualmente el agente de Message Queue como un servicio de Windows utilizando el comando `imqsvcadm`. El programa de instalación no hará por usted.
- En plataformas Windows, el cortafuegos integrado en Windows, que está activado de manera predeterminada, debe configurarse de forma manual con una regla de cortafuegos que permita al agente aceptar conexiones entrantes de clientes. (*Fallo 6675595*)
  1. En el Panel de control, haga doble clic en Firewall de Windows
 

Debe hacer clic en Continuar en el cuadro de diálogo Control de cuentas de usuario para que se abra el cuadro de diálogo Configuración de Firewall de Windows.
  2. En el cuadro de diálogo Configuración de Firewall de Windows, haga clic en la ficha Excepciones.
  3. Haga clic en Añadir programa.
  4. En el cuadro de diálogo Añadir un programa, seleccione `java.exe` y haga clic en Examinar.

Windows identifica el proceso del agente como Java Platform SE binary. Por lo tanto, busque el archivo `java.exe` que ha utilizado el agente (normalmente en `jdk1.5.0_15\jre\bin\java.exe`).

5. Haga clic en Cambiar ámbito.
  6. En el cuadro de diálogo Cambiar ámbito, seleccione “Cualquier equipo (incluyendo los que están en Internet).”
  7. Haga clic en Aceptar.
  8. En el cuadro de diálogo Añadir un programa, haga clic en Aceptar.
  9. En el cuadro de diálogo Configuración de Firewall de Windows, haga clic en Aceptar.
- En plataformas Windows, los comandos `imqadmin` y `imqobjmgr` generan un error cuando CLASSPATH contiene comillas dobles. (*Fallo 5060769*)

*Solución:* Abra una ventana de línea de comandos y anule CLASSPATH:

```
set classpath=
```

A continuación, ejecute el comando que desee con la misma ventana de línea de comandos, por ejemplo:

```
mqInstallHome\mq\bin\imqadmin
```

- La opción `-javahome` en todas las secuencias de comandos de Solaris y Windows no funciona si el valor proporcionado contiene un espacio. (*Fallo 4683029*)

Los comandos y herramientas de Message Queue utilizan la opción `javahome` para especificar un entorno de tiempo de ejecución alternativo compatible con Java 2. Pero el nombre de la ruta del entorno de tiempo de ejecución de Java alternativo no debe contener espacios. A continuación le mostramos ejemplos de rutas que incluyen espacios.

Windows: `C:\jdk 1.4`

Solaris: `/work/java 1.4`

*Solución:* Instale el tiempo de ejecución Java en una ubicación o ruta que no contenga espacios en blanco.

- El atributo `imqQueueBrowserMaxMessagesPerRetrieve` especifica el número máximo de mensajes que el tiempo de ejecución del cliente extrae a la vez cuando examina el contenido de una cola. El atributo influye en cómo se agrupan los mensajes en cola que se van a enviar al tiempo de ejecución de cliente, pero no afecta al número total de mensajes examinados. El atributo sólo afecta al mecanismo de navegación, no afecta al envío de los mensajes en cola. (*Fallo 6387631*)
- En plataformas Linux con SELinux, el comando `pkg` del Centro de actualización falla (*Bug 6892062*)

*Solución:* Este error lo provoca un problema conocido del Centro de actualización ([https://updatecenter2.dev.java.net/issues/show\\_bug.cgi?id=1211](https://updatecenter2.dev.java.net/issues/show_bug.cgi?id=1211)). Utilice el siguiente comando para activar `pkg` de modo que funcione en SELinux con activación forzada:

```
# chcon -f -t textrel_shlib_t $IMAGE/pkg/vendor-packages/openssl/crypto.so
```

## Problemas de agentes

Los problemas siguientes afectan al agente de Message Queue.

- Los clientes de Message Queue 4.4 reciben una advertencia poco clara al conectarse con agentes de Message Queue 3.7 (*Error 6899886*)

Cuando un cliente de Message Queue 4.4 se conecta a un agente de Message Queue 3.7, el cliente recibe una advertencia de tipo:

```
WARNING [I500]: Caught JVM exception: ...
[C4036]: A broker error occurred. :[505] bad version ...
```

Esta advertencia de “versión incorrecta” indica que el cliente debe conectarse al agente con un nivel de protocolo inferior.

- Al utilizar un almacén de datos JDBC, la contraseña de la base de datos se almacena como texto visible (*Error 6691717*)

*Solución:* proteger el archivo de contraseña que contiene la contraseña de la base, tal y como se describe en “[Password Files](#)” de *Sun GlassFish Message Queue 4.4 Administration Guide*.

- No se puede tener acceso al agente cuando un almacén de datos persistente abre demasiados destinos. (*Fallo 4953354*)

*Solución:* Esta situación se produce cuando el agente alcanza el límite de descriptor de archivos abiertos del sistema. En Solaris y Linux utilice la orden `ulimit` para incrementar el límite del descriptor de archivos.

- Cuando se destruye un destino, los consumidores se quedan sin referencia. (*Fallo 5060787*)

Los consumidores activos se quedan sin referencia cuando se destruye un destino. En ese caso, dejan de recibir mensajes (aunque se vuelva a crear el destino).

- Cuando un cliente JMS que utiliza servicio de conexión HTTP finaliza de forma abrupta (por ejemplo por el uso de `Ctrl-C`), el agente tarda aproximadamente un minuto en liberar la conexión del cliente y todos los recursos asociados.

Si se inicia otra instancia del cliente dentro de este periodo de un minuto y si ésta intenta utilizar el mismo ClientID, la misma suscripción duradera o cola, es posible que se genere la excepción “ID de cliente ya en uso”. No se trata de un problema real, es simplemente un efecto secundario del proceso de finalización descrito anteriormente. Si el cliente se inicia después de un retraso de aproximadamente un minuto, todo debería funcionar correctamente.

- Al utilizar la base de datos MySQL para un almacén de datos, los mensajes almacenados de más de 1 MB generarán una excepción SQL “El paquete de la consulta es demasiado grande...”. (*Fallo 6682815*)

*Solución:* Inicie el servidor MySQL con la opción `--max_allowed_packet` establecida en un valor mayor que 1 MB, el tamaño predeterminado. Por ejemplo, utilice el siguiente valor:

```
--max_allowed_packet=60M
```

- Al utilizar la base de datos MySQL para un almacén de datos compartido de alta disponibilidad, se necesita un mecanismo para configurar el motor de almacenamiento MySQL como NDBCLUSTER. (Fallo 6691394)

*Solución:* Añada la siguiente propiedad al archivo `config.properties` del agente (consulte la sección “Enhanced Clusters: JDBC Configuration Properties” de *Sun GlassFish Message Queue 4.4 Administration Guide*)

```
imq.persist.jdbc.mysql.tableoption=ENGINE=NDBCLUSTER
```

- Si utiliza el controlador 9i de Oracle (JDBC 9.2.0.x), el agente genera la excepción "Fallo al definir propiedad como persistente...". (Fallo 6626825)

*Solución:* Utilice el controlador 10g de Oracle (JDBC 10.2.0.x), para el que el agente está optimizado.

```
imq.persist.jdbc.derby.table.MYCONSTATE41.index.IDX2=
CREATE INDEX &(index) ON $(name) (MESSAGE_ID)
```

- Cuando utilice la base de datos Java DB para un almacén de datos, al almacenar un mensaje se generará la excepción SQL “no se ha podido obtener un bloqueo en el tiempo solicitado”. (Fallo 6691394)

*Solución:* Añada el siguiente valor de propiedad al archivo `config.properties` del agente::

```
imq.persist.jdbc.derby.table.MYCONSTATE41.index.IDX2=
CREATE INDEX &(index) ON $(name) (MESSAGE_ID)
```

- Al utilizar IBM JVM en AIX, el agente puede entrar en condición de memoria baja o RED sin motivo aparente (Error 6899526)

*Solución:* utilice la versión más reciente de JVM IBM (Java Runtime 1.6.0 IBM Corporation o superior) y envíe la siguiente opción de IBM JVM GC a `imqbrokerd`:

```
# imqbrokerd -vmargs -Xgcpolicy:gencon
```

## Agrupaciones de agentes

Los siguientes problemas afectan a las agrupaciones de agentes.

- Agente de alta disponibilidad con almacén de datos MySQL Cluster no puede reiniciarse si se finalizó de manera inapropiada (Error 6896877)

*Solución:* Este problema lo provoca un problema conocido de MySQL Cluster (<http://bugs.mysql.com/bug.php?id=47955>). Se ha aplicado una corrección de este problema a las versiones de MySQL 5.1.39-ndb-6.3.28, 5.1.39-ndb-7.0.9 y 5.1.39-ndb-7.1.0.

- En esta versión, sólo se admiten agrupaciones de agentes totalmente conectadas. Esto significa que todos los agentes de una agrupación deben comunicarse directamente con todos los demás agentes de la agrupación. Si se conecta a agentes de una agrupación

convencional mediante el argumento de línea de comandos `imqbrokerd -cluster`, asegúrese de que se incluyen todos los agentes del clúster.

- Si un cliente se conecta a un agente de una agrupación de agentes mejorada, el tiempo de ejecución del cliente intentará reconectar hasta que lo consiga (ignora el valor del atributo de fábrica de conexión `imqAddressListIterations`.)
- Un cliente sólo podrá examinar el contenido de las colas que estén ubicadas en su agente local. El cliente puede continuar enviando mensajes a cualquier cola o consumiendo mensajes desde una cola de cualquier agrupación; esta limitación sólo afecta a las funciones de exploración de colas.
- En una agrupación convencional que incluya agentes de la versión 4.3, todos los agentes deben ser de la versión 3.5 o posterior.
- Los agentes de Message Queue 4.3, 4.2 y 4.1 no pueden interoperar en una agrupación de forma predeterminada con agentes de Message Queue 3.7 o 3.6, dado que el valor predeterminado de `imq.autocreate.queue.maxNumActiveConsumers` se ha modificado entre estas versiones. (Fallo 6716400)

*Solución:* Asegúrese de que todos los agentes tienen el mismo valor para la opción `imq.autocreate.queue.maxNumActiveConsumers`, normalmente, puede cambiar la configuración de Message Queue 4.3, 4.2 y 4.1 para que coincida con la que se usan los agentes de las versiones 3.7 y 3.6 (cambie el valor predeterminado -1 al valor de la versión anterior: 1).

- Para añadir un agente de Message Queue 4.3 (o 4.x) a una agrupación de agentes de Message Queue 3.x, el agente principal debe estar ejecutándose. (Fallo 6763796)
- Al convertir una agrupación convencional en una mejorada, puede usar la herramienta de administración de bases de datos de Message Queue (`imqdbmgr`) para convertir un almacén de datos basado en JDBC independiente en un almacén de datos basado en JDBC compartido, como se explica en la sección “[Cluster Conversion: JDBC-Based Data Store](#)” de *Sun GlassFish Message Queue 4.4 Administration Guide*
- Un agente que utilice HADB no puede gestionar mensajes superiores a 10 MB. (Fallo 6531734)
- La conversión a un almacén HADB con el comando `imqdbmgr upgrade hasstore` puede fallar y generar el mensaje "hay demasiados bloqueos establecidos" si el almacén contiene más de 10.000 mensajes. (Fallo 6588856)

*Solución:* Utilice el siguiente comando para aumentar el número de bloqueos.

```
hadbm set NumberOfLocks=<número>
```

Para más información, consulte "Problemas de HADB" en la *Guía de resolución de problemas de Sun Java System Application Server 9.1 Enterprise Edition*.

- Si se confirman más de 500 mensajes remotos en una transacción, el agente podría devolver el error "HADB-E-12815: Espacio de memoria de tabla agotado." (Fallo 6550483)

Para más información, consulte "Problemas de HADB" en la *Guía de resolución de problemas de Sun Java System Application Server 9.1 Enterprise Edition*.

- En una agrupación de agentes, el agente coloca mensajes en la cola para una conexión remota que no se ha abierto. (*Fallo 4951010*)

*Solución:* El consumidor recibirá los mensajes cuando se abra la conexión. Los mensajes se enviarán a otro consumidor si la conexión con el consumidor sigue cerrada.

- Al consumir más de un mensaje de un agente remoto en una transacción, es posible que se registre en el agente el siguiente mensaje de error. El mensaje es benigno y puede omitirse:

```
[26/Jul/2007:13:18:27 PDT] WARNING [B2117]:
Message acknowledgement failed from
mq://129.145.130.95:7677/?instName=a&brokerSessionUID=3209681167602264320:
  ackStatus = NOT_FOUND(404)\
  Reason = Update remote transaction state to COMMITTED(6):
transaction 3534784765719091968 not found, the transaction
may have already been committed.
  AckType = MSG_CONSUMED
  MessageBrokerSession = 3209681167602264320
  TransactionID = 3534784765719091968
    SysMessageID = 8-129.145.130.95(95:fd:93:91:ec:a0)-33220-1185481094690
    ConsumerUID = 3534784765719133952\par
```

```
[26/Jul/2007:13:18:27 PDT] WARNING Notify commit transaction
[8-129.145.130.95(95:fd:93:91:ec:a0)-33220-1185481094690,
[consumer:3534784765719133952, type=NONE]]
TUID=3534784765719091968 got response:
com.sun.messaging.jmq.jmsserver.util.BrokerException:
Update remote transaction state to COMMITTED(6):
transaction 3534784765719091968 not found, the transaction may have already
been committed.:
com.sun.messaging.jmq.jmsserver.util.BrokerException: Update remote transaction
state to COMMITTED(6): transaction 3534784765719091968 not found, the transaction
may have already been committed.r
```

El mensaje se registra al notificar la confirmación al agente de inicio del mensaje para posteriores mensajes de la transacción cuando la propiedad `imq.txn.reapLimit` es baja comparada con el número de mensajes remotos de una transacción. (*Fallo 6585449*)

*Solución:* Para evitar este mensaje, aumente el valor de la propiedad `imq.txn.reapLimit`.

## Problemas con JMX

En la plataforma Windows, el método `getTransactionInfo` de `Transaction Manager Monitor MBean` devuelve información de la transacción con un tiempo de creación de transacción incorrecto. (*Fallo 6393359*)

*Solución:* Utilice el método `getTransactionInfoByID` de `Transaction Manager Monitor MBean` en su lugar.

## Compatibilidad con SOAP

Es necesario que tenga en cuenta dos cuestiones relacionadas con la compatibilidad con SOAP

- A partir de la versión 4.0 de Message Queue, ya no se admiten objetos administrados por SOAP.
- El desarrollo de SOAP depende de varios archivos: `SUNWjaf`, `SUNWjmail`, `SUNWxsrt` y `SUNWjaxp`. En la versión 4.1 de Message Queue, estos archivos sólo están disponibles si ejecuta Message Queue con JDK versión 1.6.0 o posterior.
- Anteriormente, el archivo `.jar` de la implementación de SAAJ 1.2 hacía referencia directamente a `mail.jar`. En SAAJ 1.3 ya no existe esta referencia; por lo que los clientes de Message Queue deben poner `mail.jar` explícitamente en `CLASSPATH`.

## Archivos redistribuibles

Sun GlassFish Message Queue 4.4 Update 1 contiene el siguiente conjunto de archivos que puede utilizar y distribuir libremente en formato binario:

<code>fscontext.jar</code>	<code>jaxm-api.jar</code>
<code>imq.jar</code>	<code>jms.jar</code>
<code>imqjmx.jar</code>	<code>libmqcrt.sl (HP-UX)</code>
<code>imqxm.jar</code>	<code>libmqcrt.so (UNIX)</code>
<code>imqums.war</code>	<code>mqcrt1.dll (Windows)</code>

Además, puede también redistribuir los archivos `LICENSE` y `COPYRIGHT`.

## Funciones de accesibilidad para personas con discapacidad

Para utilizar funciones de accesibilidad que han aparecido desde la publicación de este software, consulte las evaluaciones de producto de la Sección 508 (se pueden pedir a Sun) para determinar qué versiones son las mejores para utilizar soluciones accesibles. Puede encontrar las versiones actualizadas de las aplicaciones en <http://sun.com/software/javaenterprisesystem/get.html>.

Para obtener información sobre el compromiso que tiene Sun con respecto a las funciones de accesibilidad, visite <http://sun.com/access>.

## Cómo informar sobre problemas y enviar comentarios

Si experimenta problemas con Sun GlassFish Message Queue, póngase en contacto con el servicio de atención al cliente de Sun usando uno de estos procedimientos:

- Servicios de asistencia para software de Sun en línea: <http://www.sun.com/service/sunone/software>.

Este sitio incluye vínculos con la base de datos de conocimiento, el centro de asistencia en línea (Online Support Center) y la herramienta ProductTracker, así como con programas de mantenimiento y números para ponerse en contacto con el personal de asistencia técnica.

- El número de teléfono del distribuidor asociado al contrato de mantenimiento.

Para poder ayudarle lo mejor posible a resolver problemas, tenga disponible la siguiente información cuando se ponga en contacto con el servicio de asistencia:

- Descripción del problema, incluida la situación en la que éste se produce y la forma en que afecta al funcionamiento.
- Tipo de equipo, versión del sistema operativo y versión del producto, incluida cualquier revisión del producto y otro software que pudiera influir en el problema.
- Pasos detallados de los métodos que haya usado para reproducir el problema.
- Cualquier registro de errores o volcados del núcleo.

## Foro de Sun GlassFish

Puede consultar el foro de Sun GlassFish Message Queue en la ubicación siguiente:

<http://swforum.sun.com/jive/forum.jspa?forumID=24>

Apreciamos su participación.

## Foro de tecnología Java

Los foros de tecnología Java incluyen un foro dedicado a JMS que puede interesarle.

<http://forum.java.sun.com>

## Sun agradece sus comentarios

Sun tiene interés en mejorar la calidad de su documentación por lo que agradece sus comentarios y sugerencias.

Para enviar sus comentarios, acceda a <http://docs.sun.com> y haga clic en Send Comments (Enviar comentarios). En el formulario que aparece, especifique el título y el número de referencia del documento. El número de referencia es un número de siete o nueve dígitos que se encuentra en la portada del libro o en la parte superior del documento. Por ejemplo, el título de este libro es *Notas de la versión*, y el número de referencia 821-1512-10.

## Recursos adicionales de Sun

Puede encontrar información útil de Sun GlassFish en las siguientes direcciones de Internet:

- Documentación  
<http://docs.sun.com/prod/java.sys>
- Servicios profesionales  
<http://www.sun.com/service/sunps/sunone>
- Productos de software y mantenimiento  
<http://www.sun.com/software>
- Servicios de asistencia de software  
<http://www.sun.com/service/sunone/software>
- Base de conocimientos y asistencia al cliente  
<http://www.sun.com/service/support/software>
- Servicios de asistencia y formación de Sun  
<http://training.sun.com>
- Servicios profesionales y de asesoría  
<http://www.sun.com/service/sunps/sunone>
- Información para programadores  
<http://developers.sun.com>
- Servicios de asistencia para programadores de Sun  
<http://www.sun.com/developers/support>
- Formación para el uso de software  
<http://www.sun.com/software/training>

