



# Versionshinweise zu Sun Java System Message Queue 3.7 UR1



Sun Microsystems, Inc.  
4150 Network Circle  
Santa Clara, CA 95054  
U.S.A.

Teilenr.: 820-0483-10  
Februar 2007

Copyright 2007 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Alle Rechte vorbehalten.

Sun Microsystems, Inc. hat Rechte in Bezug auf geistiges Eigentum an der Technologie, die in dem in diesem Dokument beschriebenen Produkt enthalten ist. Im Besonderen und ohne Einschränkung umfassen diese Ansprüche in Bezug auf geistiges Eigentum eines oder mehrere Patente und eines oder mehrere Patente oder Anwendungen mit laufendem Patent in den USA und in anderen Ländern.

Rechte der US-Regierung – Kommerzielle Software. Für bei der Regierung beschäftigte Benutzer gelten die Standardlizenzvereinbarung von Sun Microsystems, Inc. sowie die einschlägigen Bestimmungen des FAR und seiner Ergänzungen.

Dieses Lieferung schließt möglicherweise Materialien ein, die von Fremdanbietern entwickelt wurden.

Teile dieses Produkts können von Berkeley BSD Systems abgeleitet und durch die University of California lizenziert sein. UNIX ist eine eingetragene Marke in den Vereinigten Staaten und anderen Ländern und wird ausschließlich durch die X/Open Company Ltd. lizenziert.

Sun, Sun Microsystems, das Sun-Logo, das Solaris-Logo, das Java Kaffeetassen-Logo, docs.sun.com, Java und Solaris sind Marken oder eingetragene Marken von Sun Microsystems, Inc., in den USA und anderen Ländern. Sämtliche SPARC-Marken werden unter Lizenz verwendet und sind Marken oder eingetragene Marken von SPARC International Inc. in den Vereinigten Staaten und anderen Ländern. Produkte mit der SPARC-Marke basieren auf einer von Sun Microsystems Inc. entwickelten Architektur.

Die grafischen Benutzeroberflächen von OPEN LOOK und Sun<sup>TM</sup> wurden von Sun Microsystems Inc. für seine Benutzer und Lizenznehmer entwickelt. Sun erkennt die Pionierleistung von Xerox bei der Ausarbeitung und Entwicklung des Konzepts von visuellen oder grafischen Benutzeroberflächen für die Computerindustrie an. Sun ist Inhaber einer einfachen Lizenz von Xerox für die Xerox Graphical User Interface (grafische Benutzeroberfläche von Xerox). Mit dieser Lizenz werden auch die Sun-Lizenznehmer abgedeckt, die grafische OPEN LOOK-Benutzeroberflächen implementieren und sich ansonsten an die schriftlichen Sun-Lizenzvereinbarungen halten.

Produkte, die in dieser Veröffentlichung beschrieben sind, und die in diesem Handbuch enthaltenen Informationen unterliegen den Gesetzen der US-Exportkontrolle und können den Export- oder Importgesetzen anderer Länder unterliegen. Die Verwendung im Zusammenhang mit Nuklear-, Raketen-, chemischen und biologischen Waffen, im nuklear-maritimen Bereich oder durch in diesem Bereich tätige Endbenutzer, direkt oder indirekt, ist strengstens untersagt. Der Export oder Rückexport in Länder, die einem US-Embargo unterliegen, oder an Personen und Körperschaften, die auf der US-Exportausschlussliste stehen, einschließlich (jedoch nicht beschränkt auf) der Liste nicht zulässiger Personen und speziell ausgewiesener Staatsangehöriger, ist strengstens untersagt.

DIE DOKUMENTATION WIRD "AS IS" BEREITGESTELLT, UND JEGLICHE AUSDRÜCKLICHE ODER IMPLIZITE BEDINGUNGEN, DARSTELLUNGEN UND HAFTUNG, EINSCHLIESSLICH JEGLICHER STILLSCHWEIGENDER HAFTUNG FÜR MARKTFÄHIGKEIT, EIGNUNG FÜR EINEN BESTIMMTEN ZWECK ODER NICHTÜBERTRETUNG WERDEN IM GESETZLICH ZULÄSSIGEN RAHMEN AUSDRÜCKLICH AUSGESCHLOSSEN.

# Inhalt

---

<b>1</b>	<b>Versionshinweise zu Sun Java System Message Queue 3.7 UR1</b>	<b>5</b>
	Änderungsprotokoll der Versionshinweise	6
	Grundlegendes zu Message Queue 3.7 UR1	6
	Neue Funktionen in dieser Version	6
	Hardware- und Software-Anforderungen	10
	Arbeiten mit Solaris 10-Zonen	10
	In dieser Version behobene Fehler	14
	Wichtige Informationen	15
	Installationshinweise	16
	Kompatibilitätsprobleme	16
	Documentationsaktualisierungen für Message Queue 3.7 UR1	17
	Bekannte Probleme und Beschränkungen	17
	Allgemeine Probleme	18
	Eigenschaften für das LDAP-Benutzer-Repository	18
	Broker-Cluster	19
	Administration/Konfiguration	20
	Broker-Probleme	20
	Dokumentation	22
	Dateien für die Neuverteilung	22
	Zugriffsfunktionen für Personen mit Behinderungen	23
	Problemmeldungen und Feedback	23
	Sun Java System-Software-Forum	23
	Java Technology Forum	24
	Sun freut sich über Ihre Kommentare	24
	Weitere Quellen von Sun	24



# Versionshinweise zu Sun Java System Message Queue 3.7 UR1

---

Version 3.7 UR1

Teilenr. 820-0483

Diese Versionshinweise enthalten wichtige Informationen, die zum Zeitpunkt der Herausgabe von Sun Java™ System Message Queue 3.7 UR1 zur Verfügung standen. In diesem Dokument werden neue Funktionen und Verbesserungen, bekannte Probleme und Einschränkungen beschrieben und weitere Informationen bereitgestellt. Lesen Sie dieses Dokument sorgfältig, bevor Sie Message Queue verwenden.

Die neueste Ausgabe dieser Versionshinweise werden auf der Dokumentations-Website von Sun Java System Message Queue bereitgestellt. Besuchen Sie diese Website vor der Installation und Konfiguration Ihrer Software und später regelmäßig, um stets die neuesten Versionshinweise und Produktdokumentationen verfügbar zu haben.

In diesen Versionshinweisen werden die folgenden Themen behandelt:

- „Änderungsprotokoll der Versionshinweise“ auf Seite 6
- „Grundlegendes zu Message Queue 3.7 UR1“ auf Seite 6
- „In dieser Version behobene Fehler“ auf Seite 14
- „Wichtige Informationen“ auf Seite 15
- „Bekannte Probleme und Beschränkungen“ auf Seite 17
- „Dateien für die Neuverteilung“ auf Seite 22
- „Zugriffsfunktionen für Personen mit Behinderungen“ auf Seite 23
- „Problemmeldungen und Feedback“ auf Seite 23
- „Sun freut sich über Ihre Kommentare“ auf Seite 24
- „Weitere Quellen von Sun“ auf Seite 24

In der vorliegenden Dokumentation wird auf URLs von Drittanbietern verwiesen, über die zusätzliche relevante Informationen zur Verfügung gestellt werden.

Sun ist nicht für die Verfügbarkeit von Fremd-Websites verantwortlich, die in diesem Dokument genannt werden. Sun ist nicht verantwortlich oder haftbar für die Inhalte, Werbung,

Produkte oder andere Materialien, die auf solchen Websites/Ressourcen oder über diese verfügbar sind, und unterstützt diese nicht. Sun lehnt jede Verantwortung oder Haftung für direkte oder indirekte Schäden oder Verluste ab, die durch die bzw. in Verbindung mit der Verwendung von oder der Stützung auf derartige Inhalte, Waren oder Dienstleistungen, die auf oder über diese Sites oder Ressourcen verfügbar sind, entstehen können.

## Änderungsprotokoll der Versionshinweise

TABELLE 1-1 Änderungsprotokoll

Datum	Beschreibung der Änderungen
August 2006	Erste Version dieses Dokuments.
Februar 2007	Endgültige Version dieses Dokuments.

## Grundlegendes zu Message Queue 3.7 UR1

Sun Java System Message Queue ist ein leistungsfähiger Nachrichtendienst, der ein zuverlässiges, asynchrones Messaging gemäß JMS 1.1 (Java Messaging Specification) bietet. Zusätzlich stellt Message Queue Funktionen bereit, die über die JMS-Spezifikation hinausgehen, um den Anforderungen großer Bereitstellungen im Unternehmensbereich gerecht zu werden.

Message Queue 3.7 UR1 ist ein Maintenance-Release von Message Queue 3.6, der verschiedene Fixes für die Fehlerbeseitigung sowie ein paar kleinere Verbesserungen umfasst. Der vorliegende Abschnitt umfasst die folgenden Informationen:

- [„Neue Funktionen in dieser Version“](#) auf Seite 6
- [„Hardware- und Software-Anforderungen“](#) auf Seite 10
- [„Arbeiten mit Solaris 10-Zonen“](#) auf Seite 10

## Neue Funktionen in dieser Version

Message Queue 3.7 UR1 umfasst die folgenden neuen Funktionen:

- [„Kombination der Funktionen von Platform Edition und Enterprise Edition in einer Edition“](#) auf Seite 7
- [„Änderungen an der Schnittstelle für C-API- und C-Client-Runtime“](#) auf Seite 8
- [„Formatänderungen für die persistente Speicherung“](#) auf Seite 9

Die genannten Themen werden in den folgenden Unterabschnitten näher beschrieben.

## Kombination der Funktionen von Platform Edition und Enterprise Edition in einer Edition

Zur Rationalisierung unserer Produktlinie werden die Funktionen der Platform Edition und der Enterprise Edition von Sun Java Message Queue kombiniert. Ab Message Queue 3.7 UR1 wird nur eine Edition bereitgestellt, wodurch die Funktionseinschränkungen der eigenständigen Bereitstellung wegfallen. Auf diese Weise soll die Verwendung des Produkts vereinfacht werden.

Die Kombination der verschiedenen Editionen ermöglicht ferner eine bessere Anbindung von Message Queue an das Solaris Enterprise System und bietet die Möglichkeit zur allgemeinen Nutzung der Enterprise Edition-Funktionen, jedoch ohne Anspruch auf Support, Wartung oder Haftung. Wie bei vorherigen Versionen werden weiterhin verschiedene Lizenzierungsoptionen für Support- und Wartungsdienste bereitgestellt. Message Queue wird weiterhin im Paket mit Java Enterprise System und der Application Platform Suite bereitgestellt. Bitte besuchen Sie den Sun-Onlinestore unter <http://www.sun.com>, oder wenden Sie sich an Ihren Vertriebsbeauftragten, um die optimal auf Ihre Anforderungen abgestimmte Option zu ermitteln. In der nachfolgenden Tabelle werden die Upgrade-Pfade auf die neue Einzel-Edition von Message Queue beschrieben.

TABELLE 1–2 Upgrade-Pfade für Message Queue 3.7 UR1

Vorherige Edition	Upgrade-Pfad	Kommentare
Platform Edition	Sun Java System Message Queue 3.7 UR1	Alle Funktionen (Platform und Enterprise) sind jetzt für 3.7 UR1-Kunden verfügbar. Support-Optionen stehen mit dem Erwerb dieser Lizenz zur Verfügung.
Enterprise Edition	Sun Java System Message Queue 3.7 UR1	Keine Änderungen am Funktionsumfang. Es stehen verschiedene Lizenzierungs- und Support-Optionen zur Verfügung.
Platform Edition-Support-Verträge	Upgrade auf Enterprise Edition-Support-Vertrag.	Vorhandene Support-Verträge für Vorgängerversionen der Platform Edition werden weiterhin erneuert. Für vorherige Versionen der Platform Edition werden keine neuen Verträge abgeschlossen.
Enterprise Edition-Support-Verträge	Keine Änderung.	Vorhandene Verträge werden weiterhin erneuert. Es werden weiterhin neue Verträge abgeschlossen.

In der nachstehenden Tabelle werden die Änderungen in Bezug auf die Bereitstellung für die verschiedenen Message Queue-Produkte beschrieben.

**TABELLE 1-3** Änderungen in Bezug auf die Bereitstellung von Message Queue-Produkten

Produkt	Bisherige Bereitstellungsmethode	Neue Bereitstellungsmethode	Kommentare
Open Message Queue	Nicht anwendbar	Produktseite im Sun Download Center	Eigenständiger Download. Nur Community-Support. Keine Support-Verträge verfügbar.
Message Queue Platform Edition	Message Queue-Produktseite im Sun Download Center	Nicht mehr verfügbar	Ab sofort ist nur noch die Edition von Message Queue verfügbar, die Platform- und Enterprise-Funktionen verbindet.
Message Queue Enterprise Edition-Testversion (über Platform Edition)	Message Queue-Produktseite im Sun Download Center	Testlizenz nicht länger erforderlich.	Nicht länger erforderlich.
90-Tage-Testversion der Message Queue Enterprise Edition (über Java Enterprise System-Download oder -CD)	Java Enterprise System Download Center, früher als Version 3 GA (März 2006)	Solaris Enterprise System Download Center	Solaris Enterprise System-Lizenz. Support-Optionen stehen mit der Produktlizenz zur Verfügung. (Eine 90-Tage-Testlizenz ist nicht mehr erforderlich.)
Message Queue Enterprise Edition über SunStore, CD, Einzellizenz, Java Enterprise System-Lizenz, Suite-Lizenz, bereitgestellt über Java Enterprise System	Java Enterprise System oder Suite Download Center, Medien.	Solaris Enterprise System oder Suite Download Center, Medien.	Keine Änderung.

## Änderungen an der Schnittstelle für C-API- und C-Client-Runtime

- Neue Funktion: `MQGetDestinationName()`

```
MQGetDestinationName (const MQDestinationHandle destinationHandle,
                      MQString * destinationName);
```

Verwenden Sie diese Funktion, um den Namen eines Ziels abzurufen. Der zurückgegebene Wert `destinationName` ist eine Kopie, die von der aufrufenden Komponente durch Aufruf der Funktion `MQFreeString()` freigegeben werden muss.

### Parameter

*destinationHandle* Ein Handle für das Ziel, dessen Namen Sie ermitteln möchten.  
*destinationName* Der Ausgabeparameter für den Namen.

Diese Funktion ist nützlich, wenn Sie das `ReplyTo`-Muster verwenden. Sie können mithilfe der Funktion `MQGetMessageReplyTo` ein Handle für das Ziel abrufen, an das die Nachricht gesendet werden soll. Anschließend können Sie über `MQGetDestinationName` den Namen dieses Ziels abrufen. Nach Abruf des Zielnamens können Sie die Nachrichtenverarbeitung basierend auf dem Namen durchführen.

- Neuer aufgelisteter Wert: `MQ_MESSAGE`

Der neue `MQMessageType`, `MQ_MESSAGE`, erlaubt C-Clients den Austausch von JMS-Nachrichten vom Typ `Message` mit anderen Message Queue-Clients (sowohl C als auch Java):

```
typedef enum _MQMessageType {MQ_TEXT_MESSAGE = 0,
                             MQ_BYTES_MESSAGE = 1,
                             MQ_MESSAGE = 3,
                             MQ_UNSUPPORTED_MESSAGE = 2} MQMessageType;
```

Der Typ `MQ_MESSAGE` identifiziert Nachrichten, die einen Header sowie Eigenschaften, jedoch keinen Nachrichtentext aufweisen. Sie verwenden die Funktion `MQCreateMessage()`, um eine Nachrichten von diesem Typ zu erstellen.

- Eine neue Verbindungseigenschaft, `MQ_UPDATE_RELEASE_PROPERTY`, welche die Update-Release-Version für die installierte Version von Message Queue angibt. Verwenden Sie die Funktion `MQGetMetaData()`, um Versionsinformationen abzurufen.

## Formatänderungen für die persistente Speicherung

Zur Verbesserung der Leistung wurden zwei Änderungen am Message Queue-Format für die persistente Speicherung vorgenommen. Eine Änderung betrifft den Dateispeicher, die andere den JDBC-Speicher.

- Transaktionsinformationen im Dateispeicher  
Das Format von Transaktionsstatusinformationen, die im dateibasierten persistenten Speicher von Message Queue gespeichert werden, wurde geändert, um die Datenträger-E/A zu reduzieren und die Leistung von JMS-Transaktionen zu verbessern.
- Oracle JDBC-Speicher

In vorherigen Versionen von Message Queue wurde für das mit Oracle verwendete Speicherschema der Datentyp LONG RAW zur Speicherung von Nachrichtendaten eingesetzt. Mit Oracle 8 wurde der Datentyp BLOB eingeführt, der den Typ LONG RAW ablöste. Ab Message Queue 3.7 UR1 findet ein Wechsel auf den Datentyp BLOB statt, um Leistung und Unterstützung zu verbessern.

Da sich diese Änderungen auf die Speicherkompatibilität auswirken, wurde die Speicherversion von 350 in 370 geändert. Message Queue 3.7 UR1 unterstützt eine automatische Konversion des persistenten Speichers von den älteren Versionen 200 und 350 auf 370 - sowohl für JDBC- als auch für dateibasierte Speicher. Beim ersten Start von `imqbrokerd` wird bei Ermittlung eines älteren Speichers eine Migration auf das neue Format vorgenommen. Der alte Speicher wird hierbei erhalten.

Wenn Sie dieses Upgrade rückgängig machen möchten, können Sie Message Queue 3.7 UR1 deinstallieren und die zuvor ausgeführte Version neu installieren. Da die ältere Kopie des Speichers beibehalten wird, kann der Broker mit der älteren Kopie des Speichers ausgeführt werden.

## Hardware- und Software-Anforderungen

Informationen zu den Hardware- und Software-Anforderungen von Message Queue finden Sie im *Sun Java Enterprise System Installation Guide*.

## Arbeiten mit Solaris 10-Zonen

Als *Zone* wird eine Container-Technologie von Solaris bezeichnet, die separate Umgebungen auf einem Computer bereitstellt und Anwendungen logisch voneinander isoliert. Zonen ermöglichen Ihnen die Erstellung virtueller Betriebssystemumgebungen innerhalb einer Instanz des Solaris-Betriebssystems. Durch die Ausführung von Anwendungen in unterschiedlichen Zonen können unterschiedliche Instanzen oder unterschiedliche Versionen derselben Anwendung auf einem einzigen Computer ausgeführt werden, während es gleichzeitig möglich ist, alle Ressourcen zentral zu verwalten und gemeinsam zu verwenden.

Dieser Abschnitt bietet einen kurzen Überblick über Zonen und beschreibt deren Verwendung mit Message Queue 3.7 UR1.

### Grundlegendes zu Zonen

Eine Zonenumgebung umfasst eine globale Zone und eine oder mehrere nicht globale Zonen. Wenn Solaris 10 zum ersten mal auf einem System installiert wird, ist nur eine globale Zone vorhanden. Ein Administrator kann weitere nicht globale Zonen als untergeordnete Elemente der globalen Zone erstellen. Jede Zone wird als unabhängiges System angezeigt, auf dem Solaris

ausgeführt wird. Jede Zone verfügt über eine eigene IP-Adresse, eine eigene Systemkonfiguration, eigene Instanzen der ausgeführten Anwendungen sowie einen eigenen Bereich im Dateisystem.

Die globale Zone enthält Ressourcen, die von den nicht globalen Zonen gemeinsam verwendet werden können. Auf diese Weise ist es möglich, bestimmte Verwaltungsfunktionen zu zentralisieren. Beispielsweise stehen Pakete, die in der globalen Zone installiert wurden, für alle nicht globalen Zonen zur Verfügung (Verbreitung). Dies ermöglicht eine zentrale Lebenszyklusverwaltung einschließlich Installation, Upgrade und Deinstallation. Gleichzeitig für die Isolierung durch die nicht globalen Zonen zu mehr Sicherheit und ermöglicht es Ihnen, unterschiedlich konfigurierte Instanzen oder unterschiedliche Versionen derselben Anwendung auf einem einzigen Computer auszuführen.

Bei nicht globalen Zonen wird zwischen Whole-Root-Zonen und Sparse-Root-Zonen unterschieden. Für welche dieser Zonen Sie sich als Umgebung für eine Anwendung entscheiden, richtet sich danach, welche Schwerpunkte Sie in Bezug auf administrative Steuerung und Ressourcenoptimierung setzen möchten.

- *Whole-Root-Zonen* enthalten eine Kopie des Dateisystems der globalen Zone, in der Lese- und Schreiboperationen ausgeführt werden können. In der globalen Zone installierte Pakete werden (mit Registrierungsinformationen) automatisch in die Whole-Root-Zone kopiert. Dies maximiert die administrative Steuerung, jedoch auf Kosten der Ressourcennutzung.
- *Sparse-Root-Zonen* enthalten eine Kopie eines Teils des Dateisystems der globalen Zone, in der Lese- und Schreiboperationen ausgeführt werden können. Weitere Dateisysteme werden als schreibgeschützte Dateisysteme geladen. In der globalen Zone installierte Pakete werden über schreibgeschützte Dateisysteme und über die automatische Synchronisierung von Registrierungsinformationen in den Sparse-Root-Zonen bereitgestellt. Sparse-Root-Zonen optimieren die Ressourcennutzung auf Kosten der zentralen Verwaltung.

## Java Enterprise System - Zoneneinschränkungen

Die Java Enterprise System-Komponenten hängen von einigen gemeinsam genutzten Komponenten ab, was zu einigen Einschränkungen bei der Arbeit mit Zonen führt. In einer Zonenumgebung gelten für gemeinsam genutzte Komponenten die folgenden Regeln:

- Alle gemeinsam genutzten Komponenten innerhalb einer Zone müssen dieselbe JES-Version aufweisen. Diese Anforderung hat drei Auswirkungen.
  - Wenn Sie unterschiedliche Versionen einer gemeinsam genutzten Komponente verwenden möchten, muss jede Version in einer eigenen Zone vorliegen.
  - Wenn innerhalb einer Zone eine gemeinsam genutzte Komponente auf eine neuere Version aktualisiert oder eine neuere Version installiert wird, müssen alle gemeinsam genutzten Komponenten aktualisiert werden.

- Wenn Sie gemeinsam genutzte Komponenten in der globalen Zone installieren, müssen Sie dafür Sorge tragen, dass die gemeinsam genutzten Komponenten in nicht globalen Zonen gegebenenfalls aktualisiert werden.
- Gemeinsam genutzte Komponenten können nicht in Sparse-Root-Zonen installiert werden, da das Dateisystem in Sparse-Root-Zonen lese- und schreibgeschützt ist. Stattdessen muss die Installation in der globalen Zone erfolgen. Produktkomponenten, die von gemeinsam genutzten Komponenten abhängen, müssen zunächst in der globalen Zone installiert werden und dann auf die nicht globalen Zonen verbreitet werden.

Diese Anforderungen wirken sich auf die Installation von Message Queue aus, da es sich um eine Produktkomponente von Java Enterprise System handelt und somit die Verwendung von Zonen eingeschränkt ist.

---

**Hinweis** – Message Queue wird im `/usr` Verzeichnis installiert und muss daher zunächst in der globalen Zone installiert oder aktualisiert werden.

---

## Message Queue - Fallbeispiele

Wenn Message Queue in der globalen Zone installiert ist, ist eine Weitergabe an alle nicht globalen Zonen konfiguriert. Nach der Installation von Message Queue in der globalen Zone ist in allen Zonen die gleiche Version Message Queue installiert: Melden Sie sich an einer beliebigen Zone an und führen Sie den Befehl `pkginfo -l SUNWmq` aus. Sie werden feststellen, dass in der Zone die gleiche Produktversion installiert ist wie in der globalen Zone. Anschließend können Sie unabhängige Instanzen des Message Queue-Broker in jeder Zone ausführen, da die in den Verzeichnissen `/var` und `/etc` gespeicherten Instanzen- und Konfigurationsdaten nicht gemeinsam genutzt werden. (Die meisten der weiteren Java Enterprise System-Komponenten werden nicht weitergegeben, wenn sie in der globalen Zone installiert werden.)

Da Message Queue an nicht globale Zonen weitergegeben wird, ist die globale Instanz dauerhaft an die Installation in den nicht globalen Zonen gekoppelt. Jede Deinstallation oder Aktualisierung von Message Queue in der globalen Zone wirkt sich auf die Instanzen aus, die in den nicht globalen Zonen ausgeführt werden. Das folgende Beispiel zeigt, wie dieses Verhalten zu unerwünschten Ergebnissen führen kann:

1. Sie installieren Message Queue 3.7 UR1 in der globalen Zone. Dies führt dazu, dass Message Queue 3.7 UR1-Pakete auch in allen nicht globalen Zonen installiert werden.
2. Sie deinstallieren Message Queue 3.7 UR1 in einer Whole-Root-Zone. Anschließend installieren Sie Message Queue 3.6 in der Whole-Root-Zone.

Sie verfügen nun über unterschiedliche Versionen von Message Queue in den verschiedenen Zonen. Diese Konfiguration kann in bestimmten Situationen von Vorteil sein.

3. Sie deinstallieren Message Queue 3.7 UR1 in der globalen Zone. Dies führt dazu, dass Message Queue in allen weiteren Zonen ebenfalls deinstalliert wird - einschließlich der Message Queue 3.6-Instanz in der Whole-Root-Zone.

Berücksichtigen Sie stets den kaskadierenden Effekt, den die Installation bzw. Deinstallation von Message Queue in der globalen Zone hat.

Die zwei folgenden Praxisbeispiele erläutern, wie Sie unterschiedliche Instanzen und unterschiedliche Versionen von Message Queue in verschiedenen Zonen installieren.

---

**Hinweis** – Wenn Sie Message Queue in einer Whole-Root-Zone unter Solaris 10, Solaris 10U oder Solaris 10U2 installieren möchten, müssen Sie zunächst ein Upgrade für Lockhart in der globalen Zone vornehmen. Zusätzliche Informationen finden Sie in den Umgehungsmöglichkeiten für Fehler 645030.

---

## ▼ So installieren Sie die gleiche Version von Message Queue in unterschiedlichen Zonen

- 1 **Installieren Sie die gewünschte Version von Message Queue in der globalen Zone.**

Diese Versionen werden an eine bestehende nicht globale Zone weitergegeben. Wenn Sie zusätzliche nicht globale Zonen erstellen, wird Message Queue ebenfalls an diese Zonen weitergegeben. (Sie können unterschiedliche Instanzen sowohl in Whole-Root-Zonen als auch in Sparse-Root-Zonen installieren; der Einsatz von Sparse-Root-Zonen erlaubt jedoch eine effizientere Nutzung von Speicherplatz und anderen Ressourcen.)

- 2 **Wenn Sie Message Queue an eine beliebige andere nicht globale Zone weitergeben möchten, müssen Sie diese Zonen jetzt erstellen.**
- 3 **Führen Sie eine Instanz von Message Queue in jeder nicht globalen Zone aus.**

## ▼ So installieren Sie unterschiedliche Versionen von Message Queue in verschiedenen Zonen

- 1 **Deinstallieren Sie Message Queue aus der globalen Zone.**
- 2 **Erstellen Sie Whole-Root-Zonen, und konfigurieren Sie jede Zone so, dass das Verzeichnis /usr nicht gemeinsam genutzt wird. Sie erreichen dies, indem Sie bei Erstellung der Zone die folgende Direktive verwenden:**

```
remove inherit-pkg-dir dir=/usr
```
- 3 **Installieren Sie unterschiedliche Versionen von Message Queue in jeder Whole-Root-Zone.**

**Hinweis** – Berücksichtigen Sie, dass die Installation oder Deinstallation von Message Queue in der globalen Zone Auswirkungen auf alle Instanzen (und Versionen) von Message Queue hat, die in Whole-Root-Zonen ausgeführt werden.

## In dieser Version behobene Fehler

Der Abschnitt „[In dieser Version behobene Fehler](#)“ auf Seite 14 beschreibt die in Message Queue 3.7 UR1 beseitigten Fehler.

TABELLE 1–4 In Message Queue 3.7 UR1 behobene Fehler

Fehlernummer	Beschreibung
6193884	Message Queue gibt für Gebietsschemata, die zur Nachrichtenanzeige Nicht-ASCII-Zeichen erfordern, Nachrichten mit fehlerhaften Zeichen im Systemprotokoll aus.
6251450	ConcurrentModificationException für connectList während Clusterbeendigung.
6252763	java.nio.BufferOverflowException in java.nio.HeapByteBuffer.putLong/Int.
6260076	Erste Nachrichtenveröffentlichung nach Start bei Verwendung von Oracle-Speicher erfolgt langsam.
6260814	Auswahlverarbeitung auf JMSUserID wird immer als false ausgewertet.
6264003	Der Warteschlangenbrowser zeigt Nachrichten an, die Bestandteil von Transaktionen sind, die noch nicht verarbeitet wurden.
6271876	Verbindungsdatensteuerung arbeitet nicht ordnungsgemäß, wenn ein Verbraucher mit nicht verarbeiteten Nachrichten geschlossen wird.
6284769	Speicherleck in Warteschlangenbrowser, selbst wenn ein neuer Browser erstellt und für jede Auflistung geschlossen wird.
6294767	Message Queue-Broker muss SO_REUSEADDR auf dem Netzwerksocket setzen, der geöffnet wird.
6304043	Der Broker führt keine Überprüfung durch, ob clientID für gemeinsam genutzte, nicht dauerhafte Abonnements ungleich null ist.
6307056	Das txn-Protokoll führt zu einem Leistungsengpass.
6320138	Message Queue C-API kann den Namen einer Warteschlange für einen Reply-To-Header nicht ermitteln.

TABELLE 1-4 In Message Queue 3.7 UR1 behobene Fehler (Fortsetzung)

Fehlernummer	Beschreibung
6320325	Der Broker wählt unter Solaris gelegentlich JDK 1.4 anstelle JDK 1.5, selbst wenn beide Versionen installiert sind.
6321117	Die Initialisierung eines Multibroker-Clusters führt zu einer <code>java.lang.NullPointerException</code> .
6330053	Die JMS-Client-Runtime gibt <code>java.lang.NoClassDefFoundError</code> aus, wenn eine Transaktion vom Subscriber verarbeitet wird.
6340250	Unterstützung für MESSAGE-Typ in C-API.
6351293	Hinzugefügte Unterstützung für Derby-Datenbank.
6381693	Das Flag <code>JMSRedelivered</code> wird für Nachrichten auf 'false' gesetzt, die nach einem Neustart des Verbraucher-Brokers erneut an einen Remote-Verbraucher übermittelt werden.
6388049	Bereinigung einer unvollständigen XA-Transaktion nicht möglich.
6403968	Hinzugefügtes verbraucherbasiertes Datensteuerungsprotokoll für den Lastausgleich beim Einsatz mehrerer Empfänger.
6403958	Eine Broker-Ausnahme wird ausgegeben, wenn die Warteschlange mit gelöschten Nachrichten versucht, die älteste nicht persistente Nachricht zu entfernen.
6406862	Nach dem Broker-Fehler <code>Monitor destination errormq.metrics.destination.queue.Name</code> können keine Ziele überwacht werden.
6415068	Transaktionswiederherstellung schlägt fehl für AS 8.1 UR2, generischer Ressourcenadapter für JMS Message Queue.
6421781	Verbindung mit Broker wird über folgende Syntax unterstützt: <code>-b mqtcp://hostName:portNumber/serviceName</code>
6423696	<code>Session.rollback</code> führt nicht zu Rollback verarbeiteter Nachrichten nach COMMIT REPLY-Fehler.

## Wichtige Informationen

In diesem Abschnitt finden Sie die aktuellsten Informationen, die nicht in der eigentlichen Produktdokumentation enthalten sind. In diesem Abschnitt werden folgende Themen behandelt:

- „Installationshinweise“ auf Seite 16
- „Kompatibilitätsprobleme“ auf Seite 16
- „Documentationsaktualisierungen für Message Queue 3.7 UR1“ auf Seite 17

## Installationshinweise

Refer to the *Sun Java Enterprise System Installation Guide* for information about pre-installation instructions and all other information relevant to installing the Message Queue product (bundled with Java Enterprise System) on the Solaris, Linux, and HP-UX platforms.

Detaillierte Informationen zu Upgrade und Migration auf Message Queue (als Bestandteil von Java Enterprise System) auf den Plattformen Solaris, Linux, HP-UX und Windows finden Sie im *Sun Java Enterprise System Upgrade and Migration Guide*.

## Kompatibilitätsprobleme

In diesem Abschnitt werden bekannte Kompatibilitätsprobleme in Message Queue 3.7 UR1 beschrieben.

### Schnittstellenstabilität

Message Queue verwendet zahlreiche Schnittstellen, die im Laufe der Zeit möglicherweise Änderungen unterliegen. In Anhang B, „Stability of Message Queue Interfaces“ in *Sun Java System Message Queue 3.7 UR1 Administration Guide* werden die Schnittstellen gemäß ihrer Stabilität klassifiziert. Je stabiler eine Schnittstelle, desto weniger wahrscheinlich ist eine Änderung in den nachfolgenden Produktversionen.

### Mögliche Probleme hinsichtlich der nächsten Hauptversion von Message Queue

Die nächste Hauptversion von Message Queue umfasst eventuell Änderungen, die eine Inkompatibilität mit Ihren Clients verursachen können. Wir teilen Ihnen diese Informationen jetzt mit, um Sie auf diese Änderungen vorzubereiten.

- Die Speicherorte einzelner Dateien, die als Bestandteil von Sun Java System Message Queue installiert werden, kann sich ändern. Dadurch können in vorhandenen Anwendungen Fehler auftreten, die vom aktuellen Speicherort bestimmter Message Queue-Dateien abhängen.
- Broker der Version 3.5 oder niedrigere Versionen können in einem Cluster mit neueren Brokern möglicherweise nicht mehr eingesetzt werden.
- In zukünftigen Versionen können Message Queue-Clients möglicherweise keine JDK-Versionen verwenden, die älter sind als Version 1.3.

# Documentationsaktualisierungen für Message Queue

## 3.7 UR1

In den folgenden Abschnitten werden Aktualisierungen und Änderungen an der Dokumentation zu Message Queue beschrieben. Kleinere Korrekturen und Änderungen werden hier nicht einbezogen.

### Developer's Guide for C Clients

Der *Sun Java System Message Queue 3.7 UR1 Developer's Guide for C Clients* wurde um neue Informationen zu neuen Elementen ergänzt. Es wurden Informationen zu der neuen Funktion `MQGetDestinationName`, dem neuen Nachrichtentyp `MQ_Message` sowie der neuen Verbindungseigenschaft `MQ_UPDATE_RELEASE_PROPERTY` hinzugefügt.

### Developer's Guide for Java Clients

Der *Sun Java System Message Queue 3.7 UR1 Developer's Guide for Java Clients* wurde aktualisiert. Es wurden Informationen zur Einrichtung sicherer Clients hinzugefügt, und es wird erläutert, wie Clients Ausnahmefehler verarbeiten sollten, die während der automatischen Neuverbindung auftreten.

### Administration Guide

Der *Sun Java System Message Queue 3.7 UR1 Administration Guide* wurde aktualisiert. Es wurden Informationen zu den Datenbanktabellen des Brokers hinzugefügt, und es wird erläutert, wie Sie den Broker für die Verwendung fester Ports konfigurieren.

## Bekannte Probleme und Beschränkungen

In diesem Abschnitt werden die bekanntesten Probleme mit Message Queue 3.7 UR1 aufgeführt. Dies betrifft die folgenden Produktbereiche:

- „Allgemeine Probleme“ auf Seite 18
- „Eigenschaften für das LDAP-Benutzer-Repository“ auf Seite 18
- „Broker-Cluster“ auf Seite 19
- „Administration/Konfiguration“ auf Seite 20
- „Broker-Probleme“ auf Seite 20
- „Dokumentation“ auf Seite 22

Eine Liste der aktuellen Fehler, deren Status und Umgehungsmöglichkeiten finden Sie als Mitglied der Java Developer Connection™ in der Bug Parade auf der Java Developer Connection-Website. Prüfen Sie die Informationen auf dieser Seite, bevor Sie einen neuen Fehler melden. Auch wenn nicht alle Message Queue-Fehler aufgelistet sind, ist diese Seite ein guter Ausgangspunkt, wenn Sie feststellen möchten, ob ein Problem bekannt gegeben wurde.

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

---

**Hinweis** – Die Mitgliedschaft bei der Java Developer Connection ist kostenlos, es ist jedoch eine Registrierung erforderlich. Auf der Sun Java-Webseite wird beschrieben, wie Sie Mitglied bei der Java Developer Connection werden.

---

Wenn Sie einen neuen Fehler melden oder eine Funktionsanfrage einreichen möchten, senden Sie eine E-Mail an [imq-feedback@sun.com](mailto:imq-feedback@sun.com).

## Allgemeine Probleme

- Ein Verbindungsdienst, der SSL verwendet, ist derzeit auf die Unterstützung von selbstsignierten Serverzertifikaten eingeschränkt, d.h. auf den beglaubigten Hostmodus.
- Wird ein JMS-Client bei Verwendung des HTTP-Transports plötzlich beendet (z.B. über `Strg-C`), benötigt der Broker etwa eine Minute, bevor die Clientverbindung und alle damit zusammenhängenden Ressourcen freigegeben werden.

Wird innerhalb dieses Zeitraums eine weitere Instanz des Clients gestartet, die versucht, dieselbe Client-ID, Warteschlange oder dasselbe dauerhafte Abonnement zu verwenden, wird möglicherweise ein Ausnahmefehler "Client-ID wird bereits verwendet" ausgegeben. Dies stellt jedoch kein Problem dar, es handelt sich lediglich um eine Nebenwirkung des vorangehend beschriebenen Beendigungsvorgangs. Wenn der Client nach etwa einer Minute gestartet wird, sollte kein Fehler gemeldet werden.

## Eigenschaften für das LDAP-Benutzer-Repository

In Message Queue 3.7 UR1 wird die Beispielkonfiguration für einen Broker unter Verwendung eines LDAP-Servers als Benutzer-Repository im Kommentarbereich der Datei `config.properties` angezeigt. Das Beispiel für das LDAP-Benutzer-Repository in der Datei `default.properties` ist auskommentiert.

Wenn Sie bisher einen Eigenschaftswert der Beispiel-LDAP-Benutzer-Repository-Eigenschaften in der Datei `default.properties` verwendet haben, erhält Ihr JMS-Anwendungs-Client einen Sicherheitsausnahmefehler bei dem Versuch, eine JMS-Verbindung herzustellen. Dieses Problem tritt nach einem Upgrade auf Message Queue 3.7 UR1 auf.

Wenn Ihr JMS-Client versucht, eine Verbindung mit dem Message Queue 3.7 UR1-Broker herzustellen, erhalten Sie einen Fehler im Broker-Protokoll und Ihr JMS-Client erhält folgenden Ausnahmefehler:

```

SecurityException.
20/Aug/2004:11:16:41 PDT] ERROR [B4064]: Ldap repository ldap property
.uidattr not defined for authentication type
basic:com.sun.messaging.jmq.auth.LoginException:
[B4064]: Ldap repository ldap property .uidattr not defined
for authentication type basic

```

*Umgehung* Setzen Sie die Broker-Eigenschaft `imq.user_repository.ldap.uidattr` gemäß den Anweisungen in Kapitel 7, „Managing Security“ in *Sun Java System Message Queue 3.7 URI Administration Guide*.

## Broker-Cluster

Die folgenden Punkte beziehen sich auf die Verwendung von Broker-Clustern.

- In dieser Version werden lediglich vollständig verbundene Broker-Cluster unterstützt. Dies bedeutet, dass jeder Broker in einem Cluster direkt mit allen anderen Brokern im Cluster kommunizieren muss. Wenn Sie Broker mithilfe des Befehlszeilenarguments `imqbrokerd -cluster` verbinden, stellen Sie sicher, dass alle Broker im Cluster enthalten sind.
- Ein Client, der mit einem Broker verbunden ist, der wiederum Teil eines Clusters ist, kann `QueueBrowser` nicht zum Durchsuchen von Warteschlangen nutzen, die sich auf Remote-Brokern in diesem Cluster befinden. Der Client kann nur die Warteschlangeninhalte durchsuchen, die sich auf dem Broker befinden, mit dem er direkt verbunden ist. Der Client sendet eventuell noch Meldungen an eine beliebige Warteschlange oder erhält Meldungen von einer Warteschlange oder einem Broker im Cluster. Die Einschränkung betrifft nur das Durchsuchen.
- Wird im Broker-Cluster kein Master-Broker verwendet, werden persistente Informationen, die in einem dem Cluster neu hinzugefügten Broker gespeichert sind, nicht an die anderen Broker im Cluster weitergegeben.
- Verbindung wird getrennt für einen Broker im Cluster (*Fehlernummer 6377527*).  
Dieser Fehler kann eintreten, wenn die Adresse des Brokers (dessen Verbindung getrennt wird) in die Loopback-IP-Adresse (127.0.0.1) aufgelöst wird.  
*Umgehung* Stellen Sie sicher, dass die Broker-Adresse nicht in die Loopback-IP-Adresse aufgelöst wird.
- In einem Broker-Cluster werden die Nachrichten an eine Remoteverbindung, die eventuell noch nicht gestartet wurde, in die Warteschlange gestellt (*Fehlernummer 4951010*).  
*Umgehung* Der Verbraucher erhält die Nachrichten, sobald die Verbindung gestartet wurde. Die Nachrichten werden an einen anderen Verbraucher gesendet, wenn die Verbindung beendet wird.

## Administration/Konfiguration

Die folgenden Probleme beziehen sich auf die Administration und Konfiguration von Message Queue.

- Die Dienstprogramme `imqadmin` und `imqobjmgr` geben einen Fehler aus, wenn `CLASSPATH` auf Windows-Computern doppelte Anführungszeichen enthält (*Fehlernummer 5060769*)  
*Umgehung* Sie können diese Fehlermeldung ignorieren. Der Broker informiert die Verbraucher ordnungsgemäß über mögliche Fehler. Dieser Fehler hat keine Auswirkungen auf die Zuverlässigkeit des Systems.
- Die Option `-javahome` in Solaris- und Windows-Skripts (alle Versionen) funktioniert nicht, wenn der bereitgestellte Wert ein Leerzeichen enthält (*Fehlernummer 4683029*).

Die Option `javahome` wird von den Message Queue-Befehlen und -Programmen verwendet, um eine alternative Java 2-kompatible Runtime anzugeben. Der Pfadname zur alternativen Java-Runtime darf jedoch keine Leerzeichen enthalten. Nachfolgend werden einige Beispiele für Pfade mit Leerzeichen genannt:

Windows: `C:/jdk 1.4`

Solaris: `/work/java 1.4`

*Umgehung* Installieren Sie die Java-Runtime an einem Speicherort oder unter einem Pfad, der keine Leerzeichen enthält.

- Das Attribut `imqQueueBrowserMaxMessagesPerRetrieve` legt die maximale Anzahl an Nachrichten fest, die von der Client-Runtime in einem Schritt abgerufen werden können, wenn die Inhalte eines Warteschlangenziels durchsucht werden. Beachten Sie, dass die Clientanwendung immer alle Nachrichten aus der Warteschlange abrufen. Über das Attribut `imqQueueBrowserMaxMessagesPerRetrieve` wird festgelegt, wie die in der Warteschlange enthaltenen Nachrichten aufgeteilt werden, die an die Client-Runtime gesendet werden müssen (einige große Chunks oder viele kleine Chunks), die Gesamtzahl der Nachrichten bleibt jedoch gleich. Eine Änderung des Attributwerts kann sich auf die Leistung auswirken, führt jedoch nicht dazu, dass die Clientanwendung mehr oder weniger Daten abrufen (*Fehlernummer 6387631*).

## Broker-Probleme

Die nachfolgend beschriebenen Probleme beziehen sich auf den Message Queue-Broker.

- Der `imqbrokerd -Lizenz`-Befehl zeigt veraltete und doppelte Informationen an. Auch wenn diese Lizenzart nicht mehr unterstützt wird, werden Informationen zur Testlizenz (*Fehlernummer 6489711*) und doppelte Informationen zur UNL-Lizenz (*Fehlernummer 6441015*) angezeigt.

*Umgehung* Es handelt sich um ein kosmetisches Problem, das keine Umgehung erfordert.

- Der Broker überschreitet das 1000 Nachrichten-Limit für die Warteschlange mit gelöschten Nachrichten; es werden weiterhin Nachrichten in die Warteschlange mit gelöschten Nachrichten eingefügt, bis der Broker keinen freien Speicherplatz mehr aufweist. (*Fehlernummer 6502744*)

*Umgehung* Setzen Sie das Limit für Nachrichten in der Warteschlange mit gelöschten Nachrichten auf 1001 oder einen anderen Wert als 1000.

- HTTPS `createQueueConnection` verursacht gelegentlich einen Ausnahmefehler unter Windows 2000 (*Fehlernummer 4953348*).

*Umgehung* Stellen Sie die Verbindung erneut her.

- Wenn Sie den Broker mit der Tastenkombination Strg-C beenden, werden die Transaktionen nach dem Schließen des Speichers eventuell bereinigt (*Fehlernummer 4934446*).

Der Broker zeigt eventuell Fehler mit dem Text "Speichermethodenzugriff nach Beendigung des Speichers", sofern der Broker geschlossen wird, während Nachrichten oder Transaktionen verarbeitet werden.

*Umgehung* Sie können diese Fehlermeldung ignorieren. Der Broker informiert die Verbraucher ordnungsgemäß über mögliche Fehler. Dieser Fehler hat keine Auswirkungen auf die Zuverlässigkeit des Systems.

- Wenn der Persistenzspeicher zu viele Zielstandorte öffnet, kann auf den Broker nicht mehr zugegriffen werden (*Fehlernummer 4953354*).

*Workaround* Diese Bedingung wird vom Broker verursacht, der das Deskriptor-Limit für die offenen Dateien im System erreicht. Unter Solaris und Linux erhöhen Sie das Dateideskriptor-Limit mit dem Befehl `ulimit`.

- Verbraucher verweisen, wenn ein Zielstandort gelöscht wird (*Fehlernummer 5060787*).

Aktive Verbraucher verweisen, wenn ein Zielstandort gelöscht wird. Ein verwaister Verbraucher erhält keine Meldungen mehr (auch dann nicht, wenn der Zielstandort neu erstellt wird).

*Umgehung* Derzeit gibt es keine Umgehung für dieses Problem.

- Die Nachrichtenauswahl mit `JMSMessageID` funktioniert nicht (*Fehlernummer 6196233*).

*Umgehung* Ändern Sie die Auswahl über den folgenden Ausdruck

```
JMSMessageID = "ID:message-id-string"
```

folgendermaßen ab

```
JMSMessageID IN ('ID:message-id-string', 'message-id-string')
```

- Message Queue - Ein Warteschlangen-Browser zeigt nicht verarbeitete Nachrichten (*Fehlernummer 6264003*).

Beim Durchsuchen der Inhalte einer Warteschlange werden Nachrichten, die im Rahmen einer Transaktion erzeugt, aber noch nicht verarbeitet wurden, im Warteschlangen-Browser aufgelistet.

*Umgehung* Derzeit gibt es keine Umgehung für dieses Problem.

- Wenn der Broker während der Verarbeitung abstürzt, stehen Nachrichten möglicherweise nicht mehr zur Verfügung (*Fehlernummer 6467874*).

In Ausnahmefällen stehen Nachrichten während des Broker-Absturzes für Benutzer nicht mehr zu Verfügung. Insbesondere während eines kurzen Zeitfensters im Verarbeitungsprozess kann es vorkommen, dass die Nachricht im persistenten Speicher verbleibt. Wenn dieser Fall eintritt, wird beim Start des Brokers nach einem Absturz die folgende Nachricht angezeigt.

```
[06/Sep/2006:10:11:11 PDT] ERROR [B2085]:  
  Laden von Ziel q0 [Queue] fehlgeschlagen.  
  Die auf diesem Ziel gespeicherten Nachrichten sind nicht verfügbar:  
  > com.sun.messaging.jmq.jmsserver.util.BrokerException:  
  Mit der Nachricht 8-129.145.180.87(b8:8b:26:15:41:26)-38998-1157562551217  
  ist bereits eine Bestätigungsliste verbunden.
```

*Umgehung* Derzeit gibt es keine Umgehung für dieses Problem.

## Dokumentation

Es gibt kein eigenständiges Produkt für die Beta-Version von Message Queue 3.7 UR1. Für diese Version müssen Sie Message Queue über das Installationsprogramm von Java Enterprise System Installer installieren und die Anweisungen im *Sun Java System Installation Guide* befolgen.

## Dateien für die Neuverteilung

Sun Java System Message Queue 3.7 UR1 enthält die folgenden Dateisätze, die Sie verwenden und im Binärformat verteilen können:

jms.jar	libmqcrt.so (UNIX)
imq.jar	libmqcrt.so (HPUX)
imqxm.jar	mqcrt1.dll (Windows)
fscontext.jar	

Außerdem können Sie die Dateien LICENSE und COPYRIGHT ebenfalls neu verteilen.

## Zugriffsfunktionen für Personen mit Behinderungen

Um Eingabehilfen zu erhalten, die seit der Veröffentlichung dieses Dokuments auf den Markt gekommen sind, lesen Sie Abschnitt 508 der Produktbewertungen (bei Sun auf Anfrage erhältlich), um die für Sie geeignete Version zu ermitteln. Die aktualisierten Versionen von Anwendungen finden Sie unter

<http://sun.com/software/javaenterprisesystem/get.html>.

Informationen zum Einsatz von Sun für Eingabehilfen erhalten Sie unter

<http://sun.com/access>.

## Problemmeldungen und Feedback

Wenn Sie mit Sun Java System Message Queue Probleme haben, wenden Sie sich an die Kundenunterstützung von Sun. Dazu stehen Ihnen folgende Möglichkeiten zur Verfügung:

- Online-Softwaresupport von Sun unter <http://www.sun.com/service/sunone/software>. Diese Site bietet Links zur Knowledge Base, zum Online Support Center und ProductTracker sowie zu Wartungsprogrammen und Supportkontaktnummern.
- Wenden Sie sich per Telefon an Sun. Verwenden Sie hierzu die auf Ihrem Wartungsvertrag angegebene Telefonnummer.

Damit wir Ihnen unmittelbar Hilfe anbieten können, halten Sie die folgenden Informationen bereit, wenn Sie sich an den Support wenden:

- Beschreibung des Problems, einschließlich der Situation, in der das Problem auftrat, sowie seine Auswirkungen auf Ihre Arbeit.
- Computertyp, Betriebssystem- und Produktversion, u. a. Patches und andere Softwareanwendungen, die das Problem verursacht haben könnten.
- Detaillierte Schritte zu den von Ihnen verwendeten Methoden, um das Problem zu reproduzieren.
- Sämtliche Fehlerprotokolle oder Kernspeicherauszüge.

## Sun Java System-Software-Forum

Unter der folgenden Adresse steht ein Sun Java System Message Queue-Forum zur Verfügung:

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

Wir freuen uns über Ihre Teilnahme.

## Java Technology Forum

Im Java Technology Forum finden Sie möglicherweise ein für Sie interessantes JMS-Forum.

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

## Sun freut sich über Ihre Kommentare

Sun ist stets an einer Verbesserung der eigenen Dokumentation interessiert und nimmt Ihre Kommentare und Anregungen gerne entgegen.

Sie können Ihre Kommentare unter <http://docs.sun.com> durch Klicken auf den entsprechenden Link an uns senden. Geben Sie im Online-Formular den Dokumenttitel und die Teilenummer an. Die Teilenummer ist eine 7- oder 9-stellige Zahl, die Sie auf der Titelseite des Handbuchs oder am Anfang des Dokuments finden. Der Titel des vorliegenden Buches lautet beispielsweise Versionshinweise zu Sun Java System Message Queue 3.7 UR1, die Teilenummer lautet 820-0483.

## Weitere Quellen von Sun

Nützliche Informationen über Sun Java System finden Sie unter den folgenden Internetadressen:

- Dokumentation  
<http://docs.sun.com/prod/java.sys>
- Profi-Services  
<http://www.sun.com/service/sunps/sunone>
- Softwareprodukte und Services  
<http://www.sun.com/software>
- Softwaresupport-Services  
<http://www.sun.com/service/sunone/software>
- Support und Knowledge Base  
<http://www.sun.com/service/support/software>
- Sun-Support und -Schulungen  
<http://training.sun.com>
- Beratung und Profi-Services  
<http://www.sun.com/service/sunps/sunone>
- Informationen für Entwickler

<http://developers.sun.com>

- Sun-Support-Services für Entwickler  
<http://www.sun.com/developers/support>
- Softwareschulungen  
<http://www.sun.com/software/training>

