

TIBCO Enterprise Message Service™

Release Notes

*Software Release 6.2.0
November 2011*

TIBCO provides the two-second advantage™



Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document contains confidential information that is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIBCO, The Power of Now, TIB, Information Bus, TIBCO Enterprise Message Service, TIBCO Rendezvous, TIBCO Enterprise, TIBCO SmartSockets, TIBCO ActiveMatrix BusinessWorks, and TIBCO Hawk are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

EJB, Java EE, J2EE, and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

THIS SOFTWARE MAY BE AVAILABLE ON MULTIPLE OPERATING SYSTEMS. HOWEVER, NOT ALL OPERATING SYSTEM PLATFORMS FOR A SPECIFIC SOFTWARE VERSION ARE RELEASED AT THE SAME TIME. SEE THE README FILE FOR THE AVAILABILITY OF THIS SOFTWARE VERSION ON A SPECIFIC OPERATING SYSTEM PLATFORM.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

Copyright © 1997-2011 TIBCO Software Inc. ALL RIGHTS RESERVED.

TIBCO Software Inc. Confidential Information

Contents

Preface	v
Typographical Conventions	vi
Connecting with TIBCO Resources	ix
How to Join TIBCOCommunity	ix
How to Access All TIBCO Documentation	ix
How to Contact TIBCO Support	ix
Release Notes	1
New Features	2
Compatibility With Previous Versions	8
Release 6.2.0	8
Release 6.1.0	8
Release 6.0.1	8
Release 6.0.0	8
Changes in Functionality	10
Release 6.2.0	10
Release 6.1.0	10
Release 6.0.1	11
Release 6.0.0	11
Deprecated & Obsolete Features	14
Release 6.2	14
Release 6.1	14
Release 6.0	15
Closed Issues	17
Known Issues	26

Preface

TIBCO is proud to announce the latest release of TIBCO Enterprise Message Service™. This release is the latest in a long history of TIBCO products that leverage the power of the Information Bus® to enable truly event-driven IT environments. To find out more about how TIBCO Enterprise Message Service and other TIBCO products are powered by TIB® technology, please visit us at www.tibco.com.

TIBCO Enterprise Message Service software lets application programs send and receive messages according to the Java Message Service (JMS) protocol. It also integrates with TIBCO Rendezvous and TIBCO SmartSockets messaging products.

Topics

- [Typographical Conventions, page vi](#)
- [Connecting with TIBCO Resources, page ix](#)

Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
<i>TIBCO_HOME</i> <i>ENV_HOME</i> <i>EMS_HOME</i>	<p>Many TIBCO products must be installed within the same home directory. This directory is referenced in documentation as <i>TIBCO_HOME</i>. The default value of <i>TIBCO_HOME</i> depends on the operating system. For example, on Windows systems, the default value is C:\tibco.</p> <p>Other TIBCO products are installed into an <i>installation environment</i>. Incompatible products and multiple instances of the same product are installed into different installation environments. An environment home directory is referenced in documentation as <i>ENV_HOME</i>. The default value of <i>ENV_HOME</i> depends on the operating system. For example, on Windows systems the default value is C:\tibco.</p> <p>TIBCO Enterprise Message Service installs into a directory within <i>TIBCO_HOME</i>. This directory is referenced in documentation as <i>EMS_HOME</i>. The value of <i>EMS_HOME</i> depends on the operating system. For example on Windows systems, the default value is C:\tibco\ems\6.1.</p>
code font	<p>Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example:</p> <p>Use MyCommand to start the foo process.</p>
bold code font	<p>Bold code font is used in the following ways:</p> <ul style="list-style-type: none">• In procedures, to indicate what a user types. For example: Type admin.• In large code samples, to indicate the parts of the sample that are of particular interest.• In command syntax, to indicate the default parameter for a command. For example, if no parameter is specified, MyCommand is enabled: MyCommand [enable disable]

Table 1 General Typographical Conventions (Cont'd)




Convention	Use
<i>italic font</i>	<p>Italic font is used in the following ways:</p> <ul style="list-style-type: none"> • To indicate a document title. For example: See <i>TIBCO ActiveMatrix BusinessWorks Concepts</i>. • To introduce new terms. For example: A portal page may contain several portlets. <i>Portlets</i> are mini-applications that run in a portal. • To indicate a variable in a command or code syntax that you must replace. For example: <code>MyCommand PathName</code>
Key combinations	<p>Key name separated by a plus sign indicate keys pressed simultaneously. For example: <code>Ctrl+C</code>.</p> <p>Key names separated by a comma and space indicate keys pressed one after the other. For example: <code>Esc, Ctrl+Q</code>.</p>
	The note icon indicates information that is of special interest or importance, for example, an additional action required only in certain circumstances.
	The tip icon indicates an idea that could be useful, for example, a way to apply the information provided in the current section to achieve a specific result.
	The warning icon indicates the potential for a damaging situation, for example, data loss or corruption if certain steps are taken or not taken.

Table 2 Syntax Typographical Conventions

Convention	Use
[]	<p>An optional item in a command or code syntax.</p> <p>For example:</p> <p><code>MyCommand [optional_parameter] required_parameter</code></p>
	<p>A logical OR that separates multiple items of which only one may be chosen.</p> <p>For example, you can select only one of the following parameters:</p> <p><code>MyCommand para1 param2 param3</code></p>

Table 2 Syntax Typographical Conventions

Convention	Use
{ }	<p>A logical group of items in a command. Other syntax notations may appear within each logical group.</p> <p>For example, the following command requires two parameters, which can be either the pair param1 and param2, or the pair param3 and param4.</p> <pre>MyCommand {param1 param2} {param3 param4}</pre> <p>In the next example, the command requires two parameters. The first parameter can be either param1 or param2 and the second can be either param3 or param4:</p> <pre>MyCommand {param1 param2} {param3 param4}</pre> <p>In the next example, the command can accept either two or three parameters. The first parameter must be param1. You can optionally include param2 as the second parameter. And the last parameter is either param3 or param4.</p> <pre>MyCommand param1 [param2] {param3 param4}</pre>

Connecting with TIBCO Resources

How to Join TIBCOCommunity

TIBCOCommunity is an online destination for TIBCO customers, partners, and resident experts, a place to share and access the collective experience of the TIBCO community. TIBCOCommunity offers forums, blogs, and access to a variety of resources. To register, go to <http://www.tibcommunity.com>.

How to Access All TIBCO Documentation

After you join TIBCOCommunity, you can access the documentation for all supported product versions here:

<http://docs.tibco.com/TibcoDoc>

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, please contact TIBCO Support as follows.

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:

<http://www.tibco.com/services/support>

- If you already have a valid maintenance or support contract, visit this site:

<https://support.tibco.com>

Entry to this site requires a user name and password. If you do not have a user name, you can request one.

Release Notes

This document includes release notes for TIBCO Enterprise Message Service, Software Release 6.2.0.

Check the TIBCO Product Support web site at <http://support.tibco.com> for product information that was not available at release time. Entry to this site requires a username and password. If you do not have one, you can request one. You must have a valid maintenance or support contract to use this site.

Topics

- [New Features, page 2](#)
- [Compatibility With Previous Versions, page 8](#)
- [Changes in Functionality, page 10](#)
- [Deprecated & Obsolete Features, page 14](#)
- [Closed Issues, page 17](#)
- [Known Issues, page 26](#)

New Features

This section lists features and the release they were added to this product.

Reference # and Release	Feature
New Features in Release 6.2	
6.2	<p>Listen for Client Connections Using an IPv6</p> <p>The EMS server and clients now support IPv4, IPv6 and mixed IPv4/IPv6 environments. For more information, see documentation for the <code>listen</code> parameter in the <i>TIBCO Enterprise Message Service User's Guide</i>.</p>
New Features in Release 6.1	
6.1	<p>New Platform Support</p> <p>TIBCO Enterprise Message Service is now available on:</p> <ul style="list-style-type: none">• Red Hat Enterprise Linux 6.x• Mac OS X 10.6 x86• AIX 7
6.1	<p>Destination Defrag for File-Based Stores</p> <p>A new parameter, <code>file_destination_defrag</code>, is now available in the <code>stores.conf</code> file. Destination defrag improves store file performance by batch-writing persistent pending messages from a queue to disk. This both maintains contiguous space for new messages and improves server read performance. For more information, see the <i>TIBCO Enterprise Message Service User's Guide</i>.</p>
6.1	<p>Show Messages within a Transaction</p> <p>A new administration tool command, <code>show transaction</code>, shows a list of messages that were sent or received within a specified transaction. For more information, see documentation for the <code>show transaction</code> command in the <i>TIBCO Enterprise Message Service User's Guide</i>.</p>
6.1	<p>Store Read and Write Time Statistics</p> <p>The <code>show store</code> command now gives additional information about read and write service times. For more information, see documentation for the <code>show store</code> command in the <i>TIBCO Enterprise Message Service User's Guide</i>.</p>

Reference # and Release	Feature
6.1	Selector Changes Written to Log Files <p>The EMS server now prints a selector string in the server logfile when a durable is created with a selector.</p>
6.1	Synchronize UTC Between Server and Clients <p>This release adds the ability to periodically send the EMS server's Coordinated Universal Time (UTC) time to clients, allowing the client to update its offset. This is controlled by <code>clock_sync_interval</code> in <code>tibemsd.conf</code>.</p>
6.1	Persistent Message Statistics for Destinations <p>The <code>show topic</code> and <code>show queue</code> commands now report the number and size of persistent messages. For more information, see documentation for these commands in the <i>TIBCO Enterprise Message Service User's Guide</i>.</p> <p>Additionally, four new API calls retrieve pending message count and size for persistent destinations. See documentation for these functions in the <i>TIBCO Enterprise Message Service C & COBOL Reference</i>:</p> <ul style="list-style-type: none"> • <code>tibemsQueueInfo_GetPendingPersistentMessageCount</code> • <code>tibemsQueueInfo_GetPendingPersistentMessageSize</code> • <code>tibemsTopicInfo_GetPendingPersistentMessageCount</code> • <code>tibemsTopicInfo_GetPendingPersistentMessageSize</code>
6.1	Flow Control Performance Enhancement <p>Increasing the <code>flowControl</code> limit for a queue now causes a message producer to immediately resume operation, even if the queue had already reached the limit and stalled the producer. Previously, the message producer was stalled until messages in the queue were consumed.</p>
6.1	JNDI Support for Unshared State <p>JNDI can now be used to look up connection factory objects configured to use unshared state failover.</p>
6.1	64-Bit Administration Tool <p>A 64-bit <code>tibemsadmin</code> is now available on 64-bit platforms. If both 32-bit and 64-bit tools are included with your installation, the 64-bit version is named <code>tibemsadmin64</code>. See the <i>TIBCO Enterprise Message Service Installation</i> guide for more information.</p>

Reference # and Release	Feature
6.1	Connection Factory Parameters <p>A new API call prints the parameters set in a connection factory object to a buffer. See <code>tibemsConnectionFactory_PrintToBuffer</code> in the <i>TIBCO Enterprise Message Service C & COBOL Reference</i>.</p>
6.1	Expanded FIPS Compliance <p>TIBCO Enterprise Message Service now supports FIPS compliance mode on Solaris (SPARC) platforms.</p>
New Features in Release 6.0	
6.0	Cursoring through Topics and Queues <p>It is now possible to cursor through the list returned by the <code>show topics</code> and <code>show queues</code> commands. For more information, see the command descriptions in the <i>TIBCO Enterprise Message Service User's Guide</i>.</p>
6.0	New Message Store Type <p>TIBCO Enterprise Message Service now offers an additional message store type option, called the <code>mstore</code>. This store type is designed for systems that want to store large numbers of messages in memory, without sacrificing fast recovery times in the event of a failover. For details on the <code>mstore</code> feature, see the <i>TIBCO Enterprise Message Service User's Guide</i>.</p>
6.0	Starting EMS using the Default Configuration <p>You can now quickly start the EMS server with the default configuration using scripts. The scripts <code>tibemsd.bat</code> on Windows and <code>tibemsd.sh</code> on UNIX point to the default configuration.</p>
6.0	Message Size Included in Message Tracing <p>Message tracing now includes the message size (in bytes). See the <code>msg_size</code> trace property.</p>
6.0	Host Identification in Trace Messages <p>Trace statements related to connections can identify the host by its hostname, its IP address, or both. The new <code>trace_client_host</code> option in the <code>tibemsd.conf</code> determines how hosts are identified.</p>

Reference # and Release	Feature
6.0	<p>Multicast Log Rotation</p> <p>Log rotation is now available in multicast. A <code>tibemsmcd</code> command line option enables rotation.</p> <p><code>-logfile-max-size</code> is the maximum allowed logfile size before rotation. By default, logfiles have no size limit. The minimum value for the command line paramter is 64KB.</p>
6.0	<p>Fault Tolerant Failover Notification</p> <p>New methods provide notification of each phase of the fault-tolerant failover process, including disconnection, each connect attempt, and a successful reconnection. The new methods are:</p> <p>For C</p> <ul style="list-style-type: none"> • <code>tibems_GetExceptionOnFTEvents</code> • <code>tibems_SetExceptionOnFTEvents</code> <p>For Java</p> <ul style="list-style-type: none"> • <code>com.tibco.tibjms.Tibjms.getExceptionOnFTEvents</code> • <code>com.tibco.tibjms.Tibjms.setExceptionOnFTEvents</code> <p>For .NET</p> <ul style="list-style-type: none"> • <code>TIBCO.EMS.Tibems.GetExceptionOnFTEvents</code> • <code>TIBCO.EMS.Tibems.SetExceptionOnFTEvents</code>
6.0	<p>Additional Configuration Details</p> <ul style="list-style-type: none"> • The <code>show config</code> command output now includes information from these server parameter settings: <ul style="list-style-type: none"> – <code>ssl_auth_only</code> – <code>dbstore_classpath</code> – <code>dbstore_driver_name</code> – <code>dbstore_driver_dialect</code> – <code>jre_library</code> – <code>jre_option</code> • The <code>show store</code> command in the administration tool now includes a Fragmentation field, which displays the fragmentation level for file-based stores.

Reference # and Release	Feature
6.0	Message Swapping <p>It is now possible to specify the number of messages that the TIBCO Enterprise Message Service server stores in memory before message swapping is enabled. The <code>destination_backlog_swapout</code> parameter is documented in the <i>TIBCO Enterprise Message Service User's Guide</i>.</p>
6.0	Message Recovery <p>New methods in all three APIs provide the ability to recover a message in the same way a session is recovered:</p> <ul style="list-style-type: none"> • C <code>tibemsMsg_Recover</code> • Java <code>Tibjms.recoverMsg</code> • .NET <code>Tibems.RecoverMsg</code>
6.0	epoll <p>TIBCO Enterprise Message Service now uses <code>epoll</code>, instead of <code>select()</code>, on Linux.</p>
6.0	Redelivery Delay <p>A new destination property has been added to this release of TIBCO Enterprise Message Service. The <code>redeliverydelay</code> property can be used to determine how long the EMS server waits before retuning an unacknowledged message to the message queue. For details about the new property, see the <i>TIBCO Enterprise Message Service User's Guide</i>.</p>
6.0	Incoming Message Size <p>A new parameter is available in the <code>tibemsd.conf</code> file. The <code>max_client_msg_size</code> parameter Sets a maximum size for incoming messages. When the parameter is set, the EMS server rejects incoming messages that are larger than the specified size limit.</p>
6.0	Suspend and Resume Routes <p>Two previously undocumented administration tool commands, <code>suspend route</code> and <code>resume route</code>, are now documented. For more information, see the <i>TIBCO Enterprise Message Service User's Guide</i>.</p>

Reference # and Release	Feature
6.0	Error Documentation New error code documentation, including: <ul style="list-style-type: none">• Additional Information on Transaction Exceptions When a transaction fails due to a send error or an acknowledge error, the EMS server now provides information about the operation to the client, including its status and the destination. Send errors include the inability to create the destination, lack of required ACL permissions, and an exceeded destination limit. An acknowledge error occurs when the message being acknowledged no longer exists.• Durables The Error and Status Messages appendix in the <i>TIBCO Enterprise Message Service User's Guide</i> now includes details about the warning message: Deleting and recreating durable <durable name> due to change in client attributes: destination.• Temporary Destinations The EMS server log now includes additional information about the message ERROR: Invalid request to delete temporary destination. The new information includes the connection ID, making it easier to determine which client is sending bad requests.

Compatibility With Previous Versions

This section explains how to migrate from a previous release to this release.

Release 6.2.0

There are no migration procedures or compatibility issues in this release.

Release 6.1.0

There are no migration procedures or compatibility issues in this release.

Release 6.0.1

Multicast Applications

This release of TIBCO Enterprise Message Service uses UDP encapsulation rather than raw sockets when broadcasting multicast data. Applications using multicast messaging must perform the following steps to update the messaging application:

1. Shut down the EMS server.
2. Shut down the multicast daemon.
3. Replace the server and multicast daemon binaries.
4. Restart the multicast daemon.
5. Restart the EMS server.

EMS clients will automatically update, and begin using UDP encapsulation. See [Multicast UDP Encapsulation on page 11](#) for more information.

Release 6.0.0

Updating the Database Schema

The 6.0 release of TIBCO Enterprise Message Service includes some enhancements and changes to the database store feature. After installing the new version of EMS, you must run the EMS Schema Export Tool with the **-updateall** **-export** options to apply these changes to your database store implementation.

Browsing Queues with an Older EMS Client

For best performance, TIBCO Enterprise Message Service clients should upgrade to release 6.0.0 if they consume messages from queues that store messages in mstores. That is, a queue where the destination store property is set to a store of type mstore. If a 5.x or below client browses an mstore-based queue with an EMS 6.0.0 server, the server prints a warning that the client browsing the queue must be upgraded for optimum performance.

Synchronization with TIBCO Enterprise Message Service Central Administration

TIBCO Enterprise Message Service deployments that include mstore feature parameters are not supported by TIBCO Enterprise Message Service Central Administration version 1.0.

Failsafe

The `failsafe` parameter is deprecated and is not compatible with the multiple store feature. The same functionality can be achieved through the configuration of message stores and default store files.

When upgrading, update `queues.conf` and `topics.conf` to replace the `failsafe` property with:

```
store= $sys.failsafe
```

Failure to do so results in a mixed configuration error.

Transactional Consumers on Routed Queues

TIBCO Enterprise Message Service does not support consuming messages transactionally from a routed queue. However, previous versions of EMS did not prevent applications from doing so. Starting in EMS 6.0, message consumption from a routed queue is rejected. If you have existing applications which depend on this behavior, you may use the

`allow_unsupported_tx_routed_q_consumers` parameter in the `tibemsd.conf` to allow these consumers.

Changes in Functionality

This section lists changes to product functionality and the release when the change was introduced.

Release 6.2.0

There are no changes in functionality in this release.

Release 6.1.0

The following are changes in functionality in this release.

- **Installation Batch Scripts** Installation batch scripts (*.bat) on Windows must now invoke the TIBCO Universaller Installer using the CALL command. Scripts that invoke the installer directly must be changed to use the CALL command:

```
CALL .\TIBCOUniversalInstaller -silent
```
- **Root Access for Multicast** root access is no longer required to run the multicast daemon.
- **Destination Backlog Limit Enforced** The `destination_backlog_swapout` setting is now implemented on destinations with no current consumers. Previously, the server limited the messages in memory to 8MB for a destination with no current consumers, regardless of the `destination_backlog_swapout` setting. The server now honors the limit set by `destination_backlog_swapout` even when there are no consumers.
- **getStoreInfo method** Store names should now be looked up by store name instead of type. The Java method `StoreInfo getStoreInfo(int type)` in the `TibjmsAdmin` class is deprecated, along with the related .NET method. Use instead the new method:

```
StoreInfo getStoreInfo(String name)
```
- **Bridge atomicity** When message limits set by `max_msg_memory`, `max_msgs`, or `max_bytes` parameters are reached, the entire bridge operation fails, regardless of the settings on different bridge targets. Previously, part of an operation could succeed while the rest failed.
- **Start-up fails if LDAP connection is faulty** During start-up, the EMS server checks for the existence of specified files and attempts to bind with the LDAP server. If the EMS server is able to connect to the LDAP server but encounters an error during binding, startup fails.

- **JAAS and JACI classloader** The `jaci_classpath` and `jaas_classpath` parameters now employ the same classloader, allowing variables to be shared between the modules. This change will only be apparent if there was an error in your previous installation.

Release 6.0.1

Multicast UDP Encapsulation

In previous releases of TIBCO Enterprise Message Service, the EMS server used raw sockets when in multicast mode. Raw sockets are no longer supported. The server uses UDP encapsulation when sending multicast data.

For instruction on updating your multicast application, see [Multicast Applications on page 8](#).

Server Memory Use

The EMS server now rejects message publish requests almost immediately after reaching the `max_msg_memory` limit. This prevents the server from draining system resources, and decreases CPU usage. When message memory usage drops, the server accepts the publish requests.

Release 6.0.0

Close Queue Browsers

You *must* close the queue browser after an application finishes with the browser. An open queue browser maintains its state on the EMS server, and can affect server operations and statistics.

.NET Assembly Versioning

This release of TIBCO Enterprise Message Service introduces versioning in the .NET assembly files. Prior to TIBCO Enterprise Message Service release 6.0.0, all EMS .NET assemblies showed an assembly version number `1.0.0.0`, which allowed client applications to upgrade to the latest version of EMS without rebuilding.

This same functionality is now available through the introduction of `policy` DLL files, which are included for each EMS assembly.

Enabling SSL in Clients

To use an SSL connection to the EMS server, a client must now include these JAR files in the CLASSPATH:

- `tibcrypt.jar`
- `slf4j-api-1.4.2.jar`
- `slf4j-simple-1.4.2.jar`

These JAR files are installed in `EMS_HOME/lib`.

Deleting Topics using the Administration Tool and API

The administration tool and admin API cannot delete a topic if matching messages are awaiting delivery to a consumer on a different topic. An administrative deletion of specific topics will now fail if some of the matching messages could not be discarded because they are pending for consumer(s) on a different topic.

Automatic Message Buffer Resizing

The TIBCO Enterprise Message Service server now automatically decreases the size of its internal message buffer when the recent incoming messages significantly smaller than the buffer size. The buffer shrinks only in response to incoming messages or pings. Therefore, if an application stops sending messages and `client_heartbeat_server` is not set, this shrinkage does not occur.

Dynamic Routed Queues

As in previous releases, dynamic routed queues are illegal. However, the server now prints a message to this effect. If the `queues.conf` file includes a dynamic queue with the global property set to enable routing, the TIBCO Enterprise Message Service server now prints a message that dynamic routed queues are not supported.

Exception Thrown when Route Undefined

When creating a durable, the administration tool and admin API throw an exception if the `route` parameter is `true` and a route has not been defined. Previously, no exception was thrown when a route was not defined.

Send and Receive Buffer Sizes

On Linux platforms, the EMS server does not by default set the send and receive buffer sizes. By default, Linux auto-tuning controls buffering. To control this behavior, set the `socket_send_buffer_size` and `socket_receive_buffer_size` parameters in the `tibemsd.conf`.

Destination Statistics Header File

The destination statistics header file has been renamed from `ddeststat.h` to `deststat.h`. The file is located in:

```
./include/tibems/deststat.h
```

Method Name Change

In previous releases the method name for `getExlicitConfigOnly` included a typo. The method is now correctly named `getExplicitConfigOnly`.

PRODCONS Tracing Scope

The server trace item `PRODCONS` now traces the initialization of queue browser, in addition to printing a message when a client creates or closes a producer or consumer.

Support for TIBCO Rendezvous

This release of TIBCO Enterprise Message Service integrates with TIBCO Rendezvous Software Release 8.2.2. Rendezvous users are strongly encouraged to install this upgrade release on all hardware.

Support for TIBCO SmartSockets

This release of TIBCO Enterprise Message Service integrates with TIBCO SmartSockets Software Release 6.8.2. SmartSockets users are strongly encouraged to install this upgrade release on all hardware.

Deprecated & Obsolete Features

This section describes deprecated features (if any), and lists features, if relevant, that may be useful alternatives to the deprecated features. Any use of a deprecated feature should be discontinued as it may be removed in a future release. You should avoid becoming dependent on deprecated features and become familiar with the suggested alternative features.

This section also lists features that are removed (if any).

Release 6.2

No features are deprecated or removed in this release.

Release 6.1

Platform Support

Please note the following changes in platform support.

Platform	Status	Description
Mac OS X 10.4 and 10.5 PowerPC	Obsolete	As of this release, only Mac Intel hardware is supported.
Mac OS X 10.5 x86	Deprecated	TIBCO Enterprise Message Service now supports Mac OS X 10.6.
OpenVMS Alpha	Deprecated	OpenVMS Itanium is supported.
Solaris 8, 9 Sun SPARC	Deprecated	TIBCO Enterprise Message Service now supports Solaris 10 (SPARC).

Deprecated Features

The following features are deprecated in this release.

- **Single-Threaded Applications on VMS** Support for single-threaded applications on OpenVMS platforms is deprecated.
- **Java Developer Kit (JDK), version 1.5** Support for the JDK 1.5 will be removed in the next release. Users should prepare to migrate to JDK 1.6.

- **getStoreInfo method** The Java method `StoreInfo getStoreInfo(int type)` in the `TibjmsAdmin` class is deprecated, along with the related .NET method. Store names should now be looked up by store name instead of type, using the method:
`StoreInfo getStoreInfo(string-name)`

Removed Features

The following features were removed in this release.

- The *TIBCO Enterprise Message Service Application Integration Guide* has been removed from this release.

Release 6.0

Deprecated

The following admin APIs are deprecated and will be removed in a future release:

- **Java** `ServerInfo.getFaultTolerantFailoverReread`
- **.NET** `ServerInfo.FaultTolerantFailoverReread`

.NET Compact Framework

Support for the .NET Compact Framework is deprecated. In a future release, support for the .NET Compact Framework library will be removed.

Document Deprecated

The *TIBCO Enterprise Message Service Application Integration Guide* is deprecated. Future releases of TIBCO Enterprise Message Service will not include this guide.

Platform Support

Please note the following changes in platform support.

Platform	Status	Description
AIX 5.2	Obsolete	TIBCO Enterprise Message Service now supports AIX 5.3 and later.

Platform	Status	Description
Mac OS X 10.4 x86	Obsolete	TIBCO Enterprise Message Service now supports Mac OS X 10.5 and later.
Mac OS X 10.4 PowerPC	Deprecated	TIBCO Enterprise Message Service now supports Mac OS X 10.5 and later.
Mac OS X 10.5 PowerPC	Deprecated	Additionally, TIBCO plans to support only Mac Intel hardware in a future release.

Closed Issues

This section lists issues that were closed in the named releases. Unless otherwise noted, the closed issue affected installations across all platforms.

Reference #	Description
Issues Closed in Release 6.2.0	
There are no closed issues to report in this release.	
Issues Closed in Release 6.1.0	
EMS-3851	Fixed an issue that could cause the EMS server to crash when the client connected and disconnected frequently.
EMS-3823	Fixed a problem that prevented temporary reply to queues from being removed after messages were routed to the routing partner.
EMS-3810	Fixed a problem that prevented .NET clients from recovering prepared transactions after an EMS server fault tolerant failover.
EMS-3760	Fixed an issue that caused the log file to rotate at only half the size specified for <code>logfile_max_size</code> in <code>tibemsd.conf</code> .
EMS-3755	Fixed an error related to EMS server recovery after the server ran out of space in the file system while processing a large transaction. Previously, after restarting the <code>tibemsd</code> recovered messages that should not have been recovered. This has been corrected.
EMS-3737	Previously, an EMS client using a transacted session did not always get an exception even though no messages were saved in the server. This could occur when the server failed writing messages to <code>sync-msgs.db</code> . This has been corrected; the client now gets an exception noting that no messages were written to the server.
EMS-3730	Previously, the factory print method <code>toString()</code> did not indicate whether the factory used unshared state failover (UFO). The method now reports whether the factory is a UFO connection factory.
EMS-3729	Fixed a problem that prevented EMS clients from failing when the EMS server failed to extend the used portion of the store file. This only affects file-based stores.

Reference #	Description
EMS-3705	Fixed an error that allowed the EMS server to accept a <code>max_client_msg_size</code> setting greater than its maximum permitted setting. The server now ignores values greater than or equal to 2GB.
EMS-3699	Fixed an issue that sometimes caused the EMS server to crash after compacting and then attempting to update a database store file. This occurred when the EMS server did not have the privileges needed to access the store files.
EMS-3695	Fixed an issue that could cause some messages to reappear in the queue following a purge. This could occur when a queue had multiple consumers with unacknowledged messages at the time of purge.
EMS-3691 EMS-3690	Modified the behavior of the <code>destination_backlog_swapout</code> setting from being implemented on a destination with no current consumers. Previously, the server limited the messages in memory to 8MB for a destination with no current consumers, regardless of the <code>destination_backlog_swapout</code> setting. The server now honors the limit set by <code>destination_backlog_swapout</code> even when there are no consumers.
EMS-3687	Fixed an issue that was preventing passive routes from being removed when the active server disconnected. The passive server would also incorrectly maintain system durable subscribers for the active route.
EMS-3667	Fixed an error that could cause the EMS server to crash when an SSL certificate using PKCS #7 DER encoding was specified for <code>ssl_server_trusted</code> .
EMS-3664	Fixed a problem on Windows systems that prevented an mstore file from being created if its absolute path was specified instead of its relative path.
EMS-3662	Fixed an error that caused the Synchronous Storage size displayed by the <code>show server</code> and <code>info</code> commands to display an incorrect value.
EMS-3658	Corrected an error that sometimes prevented the prefetch value from being set successfully using the <code>addprop</code> command.
EMS-3657	Fixed an issue that prevented some messages from being discarded following a queue purge operation. This problem affected messages pending for consumers that were closed but not disconnected prior to the purge.
EMS-3656	Fixed an error that caused messages to disappear from the <code>show queue queue-name</code> results after the queue consumer was closed.
EMS-3651	Fixed an issue that could cause a permission denied error despite a successful installation.

Reference #	Description
EMS-3635	Fixed an error that could sometimes result in incorrect values being reported for admin commands. The presence of the <code>sender_name</code> property on a destination could cause a negative size reported in several admin command outputs (such as Pending Message Size in <code>info</code> command, Pending messages sent in <code>show consumers</code> , and so on).
EMS-3632	Removed two invalid parameters: the <code>multicast_udp_encapsulation</code> parameter was removed from <code>tibemsd.conf</code> , and <code>-udp</code> parameter was removed from the <code>tibemsmcd</code> daemon. UDP encapsulation is now standard in EMS.
EMS-3628	Fixed an issue that could cause consumers on temporary destinations to silently stop receiving messages if the client application reconnected to the server after a network issue. With <code>PRODCONS</code> tracing enabled, the following trace would be seen: [<code><unknown>@hostname</code>]: Destroyed consumer.
EMS-3620	Temporary destinations were not always cleaned up periodically as they should have been. This has been fixed.
EMS-3617	Fixed an issue that sometimes prevented an EMS route from being reestablished following a network switch outage.
EMS-3616	Improved exception handling for EMS .NET applications. Previously, the application could return misleading errors.
EMS-3614	Messages that were imported from Rendezvous, SmartSockets and RVCN transports did not have the <code>JMSTimestamp</code> header set. The behavior of transports has changed. On all transports, messages without a timestamp set will have the <code>JMSTimestamp</code> header set at time of import.
EMS-3582	Fixed an issue that caused the EMS server to send multiple protocol messages to all connected routed servers following one server shutdown.
EMS-3576	Fixed an error that caused a message to be received multiple times from the same durable subscriber between different sessions.
EMS-3561	EMS server could crash when it sends a topic interest protocol message to another server just as the connection to the other server is broken. This has been fixed.
EMS-3556	Fixed an error that allowed the server to start even if invalid destination or parameter settings were specified in the <code>topics.conf</code> file. The server now correctly stops when invalid settings exist.

Reference #	Description
EMS-3552	Fixed an error that could cause the C library to prematurely trigger a timeout. This problem surfaced when a connection attempt timeout of less than 1000 was specified.
EMS-3551	Fixed an error that sometimes allowed multiple queue consumers to receive and acknowledge the same message. This problem occurred when a fault tolerant consumer disconnected after preparing an XA transaction.
EMS-3125	Fixed an issue that prevented SSL keys from being stored in a RACF database.
EMS-3005	The tibemsadmin tool incorrectly accepted a 'routed' property when creating or modifying a queue. This has been fixed.
EMS-2808	Previously, the EMS trace log for compact store events could incorrectly identify a synchronous store as asynchronous. This has been fixed. The trace about compacting a store no longer mentions whether the store is synchronous or asynchronous.
EMS-2781	Fixed an issue that sometimes caused messages to be stored in memory rather than the database store, without returning an error. This could occur when the disk was full and messages were published to a topic and bridged to a queue.
EMS-2665	Fixed an error that could sometimes cause duplicate messages on queues bridged from a global topic.
EMS-2416	Previously, the Java Admin API could not retrieve the database store classpath. This has been corrected with the addition of five new methods to the ServerInfo class.
EMS-2135	Fixed an error that could cause a deadlock when multiple destinations were dynamically switched to a different store at the same time, and while transactional publications were occurring. Messages are now written to the store that their destination was pointing to at the time of the commit/prepare, even if that store was switched immediately after the initial commit/prepare.
1-B9QC6R (EMS-2612)	Fixed a problem that prevented the EMS server from destroying the queue receivers on a global queue when those receivers connected to a routed server.

Issues Closed in Release 6.0.1

EMS-3612	The EMS server encountered problems when privileges were lowered. This occurred in multicast applications where raw sockets were used. By default, the server now uses UDP encapsulation. Raw sockets are no longer supported.
----------	--

Reference #	Description
EMS-3610	EMS server could crash when trying to administratively destroy destinations with active consumers. This has been fixed.
EMS-3608	Creating certain classes from object messages failed when used with the WebLogic application server. This has been fixed
EMS-3606	Fixed an error that could cause the server to crash if a destination being destroyed contained closed consumers with outstanding unacknowledged messages.
EMS-3605	Fixed an error that resulted in an inaccurate JMSRedelivered flag setting. This error was seen when messages on a topic were redelivered following an XA rollback.
EMS-3599	Fixed a problem that caused the EMS server to crash when processing incoming data messages and creating monitor messages for route connect errors concurrently.
EMS-3596	If a session with a closed consumer that had unacknowledged messages was disconnected, the server could crash. This has been fixed.
EMS-3593	EMS Server's <code>tibemsadmin info</code> command sometimes displayed an inaccurate "Routed Msg Hold Count" count. Non-persistent messages arriving from a route could cause the count to increment incorrectly. This has been fixed.
EMS-3592	<p>The EMS server had high CPU usage after the EMS server reached its <code>max_msg_memory</code> limit. This was true as long as messages kept coming in to a global topic which was bridged to queues.</p> <p>This has been fixed. When <code>max_msg_memory</code> is reached, publishes now fail almost immediately, preventing the EMS server from draining system resources. When message memory usage drops, the server accepts the publish requests.</p>
EMS-3590	Fixed an error that sometimes caused an EMS java client to hang when attempting to close a session. The hang occurred after closing the message consumer in an OnMessage callback function with <code>Tibjms.SetAllocCloseInCallback</code> set to <code>true</code> .
EMS-3584	Fixed an error where a message stream could stall if a routed queue consumer attempted to recover a session or rollback a transaction while the server that owned the queue was down.

Reference #	Description
EMS-3580	<p>Fixed an error in the way large message bodies are truncated in the trace file. The EMS server prints up to one kilobyte of a message string field, with a total message size of 8 KB. The trace message now indicates if the full message is not printed.</p> <p>Note that this behavior is incorrectly described in the EMS documentation, and will be corrected in the next release of the EMS documentation.</p>
EMS-3579	Fixed an error that caused the <code>msg_pool_size</code> parameter to allocate memory incorrectly.
EMS-3575	Fixed an error that caused a memory leak when creating and closing an EMS SSL connection.
EMS-3566	Fixed an error that could cause integer overflows when the <code>TibjmsxStream.readString(bool)</code> method was used.
EMS-3562	Fixed an error that caused the EMS service name to display incorrectly on Windows systems following a server restart. This problem was seen when the EMS service was registered using a suffix.
EMS-3173	<p>Previously, the following was not explicitly set in the <code>datastore.properties</code> file:</p> <pre>Dhibernate.connection.provider_class=org.hibernate.connection.C3PO ConnectionProvider</pre> <p>This has been fixed.</p>

Issues Closed in Release 6.0.0

1-A6Y7Y7	Previously, the administration tool did not verify permissions when creating a JNDI name for a topic or queue using the command <code>create jndiname</code> . This has been corrected, and permissions are now checked.
1-A5PK2T	Fixed an error that allowed the <code>create factory</code> command in the admin tool to create a connection factory without specifying the URL. The URL is a required field, and the command fails if a valid URL is not specified.
1-AX869K	While processing the valid message record, the EMS server encountered an invalid sequence of bytes which did not represent a properly serialized sequence number. The current shipping EMS servers cannot progress beyond this corruption.
1-AEEYNQ	Fixed defect where messages were getting automatically swapped out of memory and into store upon recovery when <code>msg_swapping</code> was disabled.

Reference #	Description
1-AVDWRD	Fixed an error that sometimes caused the EMS server to crash when purging a queue with pending messages after running into reserve mode.
1-AKFFE4	Fixed an error in EMSSSLSystemStoreInfo that sometimes caused the EMS client to receive an error when attempting to create an SSL connection to the server.
1-ABTU97	Fixed an error that sometimes displayed the wrong stack trace for tibemsErrorContext object.
1-A2W5ZG	In infrequent cases, the EMS server sometimes crashed when receiving a duplicate "session commit" request from the same session. This has been fixed.
1-9GB3ND	Previously, the EMS server Windows service name included a space between tibemsd and the suffix. This has been corrected. The emsntsrsg.exe binary now places an underscore in the service name between the name and the suffix, rather than a space.
1-9CDF69	Previously, the administration tool did not allow you to set the multicast statistics interval, although the admin API did. This has been corrected to allow the interval to be set using the tibemsadmin tool. Additionally, the user is now notified when the value of the specified multicast statistics interval is below minimum allowed.
1-9HWO5B	Fixed an error that caused delayed message acknowledgements for routed messages. In addition to an improved handling of acknowledgements, a new route trace message is now printed just prior to connecting to the remote server. If the connection blocks, it is possible to determine which remote server connection failed.
1-9DVOH5	Fixed an error that sometimes caused duplicate messages in routed configurations. This problem occurred when a durable subscription for a route was administratively created before the route itself was created. The server now returns an error if a route durable is created for a route that doesn't exist.
1-927B8P	In a routed configuration, if the downstream server has track_message_ids enabled, it would sometimes discard a resent message as a duplicate. This occurred following a consumer disconnection, where another consumer is waiting on the queue. This has been fixed.
1-86TV25	Fixed an error that sometimes caused C clients to crash when SSLParams were not set and the client passed an SSL server URL in either tibemsConnection_Create or tibemsConnectionFactory_CreateConnection. The C client now rejects this combination as invalid.

Reference #	Description
1-AL7N3H	Fixed an error that sometimes caused the EMS server to abort during startup. This occurred when the <code>ft_reconnect_timeout</code> was greater than 300, and the <code>startup_abort_list</code> was set to <code>CONFIG_ERRORS</code> .
1-8SDX4E	Fixed a bug where, although queue names of up to 249 characters are allowed in <code>queues.conf</code> , a queue name longer than 121 characters in <code>acl.conf</code> would cause the server to report an error during startup.
1-AHE8EB	Fixed an error that sometimes caused handle leakage when EMS connections were created or closed using the EMS .NET DLL and .NET Admin DLL.
1-9T9Y3I	Fixed an error that sometimes caused .NET clients to throw unhandled null reference exceptions when reconnection was triggered and <code>connectAttemptTimeout</code> or <code>ReconnectAttemptTimeout</code> was used.
1-9KF4B7	Fixed an error that sometimes caused errors in fault tolerant scenarios where database stores are used. The errors reported were <code>Dual-Active server detected</code> and <code>Cannot refresh when not locked</code> .
1-AOTGJH	Fixed an error in message recovery that permitted <code>acknowledge()</code> to be called on a message that was already recovered.
1-ACRFQX	Fixed an error that allowed C clients to fail to report errors returned by the compression library and instead send messages without the JMS Body. This has been fixed.
1-A3TU3B	Fixed an issue, present since EMS 5.0, that would prevent ACLs on system queues (such as <code>\$sys.undelivered</code>) if no parent queue (such as <code>></code>) was present in <code>queues.conf</code> .
1-989RVU	Fixed an error that caused the <code>ServerInfo</code> to print unknown <code>user_auth</code> setting when <code>jass</code> was specified.
1-8YJLG2	Fixed a problem with the <code>compact</code> command that caused the server to ignore a specified timeout, if a space character was included after the timeout value.
1-7Z926S	Fixed a problem where the <code>tibemsadmin create user</code> command, when given a password that is too long, creates the user with no password. Passwords can now be up to 4096 characters.
1-AV189G	This resolves an issue that prevented the EMS server from reconnecting to the LDAP server after the LDAP server was reset.

Reference #	Description
1-AGWO0P	Fixed an error that sometimes caused the EMS server crash after failing to allocate sufficient memory needed to process a message.
1-9WBE7N	Previously, the size limit for including message bodies in the trace file was undocumented. This has been corrected. Setting <code>trace=body</code> includes the message body in trace messages when the message size is less than 1 MB, including the header. When a message is larger than 1 MB, the body is not included in trace messages.
1-9ZDT3A	Previously, the warning message <code>Deleting and recreating durable due to change in client attributes</code> was not clearly printed as a warning. This has been corrected.

Known Issues

The table in this section lists known issues in this release.

Identified in Release	Summary/Workaround
6.1.0	<p>Summary The JMSHeader fields of messages imported from SmartSockets cannot be modified.</p> <p>Workaround None.</p>
6.0.0	<p>Summary Using both multicast and the SmartSockets bridge at the same time is not supported.</p> <p>Workaround None.</p>
6.0.0	<p>Summary The EMS server does not load OCI drivers (used with the OracleRAC database server).</p> <p>Workaround In order to load the OCI libraries, specify the driver location using the <code>module_path</code> parameter in the <code>tibemsd.conf</code>. For example:</p> <pre>module_path=/rv/tools/tibjms/Oracle11gClient/linux24gl23/x86/oci32</pre> <p>Note that TIBCO SmartSockets users also use the <code>module_path</code> parameter to dynamically load the SmartSockets library files. In order to define both OCI and SmartSockets library locations, separators should follow the same conventions used to specify <code>PATH</code>. On Unix platforms separate paths using a colon (:). On Windows platforms, use a semicolon. For example:</p> <pre>module_path= c:\tibco\ss\bin\i86_w32;c:\Oracle11gClient\oci32</pre>
5.1.5	<p>Summary When running EMS in FIPS compliant mode, DSA certificates cannot be used.</p> <p>Workaround Use RSA certificates when running EMS in FIPS compliant mode.</p>
5.1.5	<p>Summary On HP-UX 11.11, neither the OpenLDAP libraries (static or dynamic) nor the EMS server support SSL connections to an LDAP server.</p> <p>Workaround Connect to the LDAP server using a non-SSL connection.</p>
5.1.4	<p>Summary The OpenLDAP shared libraries for HP-UX 11.11 do not support SSL connections to an LDAP server. Non-SSL connections are supported.</p> <p>Workaround When SSL connections are required, use the OpenLDAP static libraries.</p>

Identified in Release	Summary/Workaround
5.1.4 1-AC2L2T	<p>Summary On HP/UX platforms, authentication with LDAP can fail in some situations when the <code>ldap_operation_timeout</code> parameter is set.</p> <p>Workaround If failure occurs, do not use <code>ldap_operation_timeout</code>.</p>
5.1.1	<p>Summary Documentation for the unshared state failover feature in the .NET client is not available through the standard TIBCO Enterprise Message Service documentation interface.</p> <p>Workaround To access the documentation, open the <code>N_TIBCO_EMS_UFO.htm</code> file located in:</p> <p><code>EMS_HOME\doc\html\tib_ems_api_reference\api\dotNETUFO\html</code></p>
5.1 1-94OBDW	<p>Summary Some clients may encounter errors when receiving messages with large Correlation IDs.</p> <p>Workaround The JMS Correlation ID should be limited to 4 KB in size.</p>
5.0	<p>Summary The SmartSockets bridge is not supported for the 64-bit EMS server on the following platforms:</p> <ul style="list-style-type: none"> • <code>hpux111/hppa</code> • <code>hpux112/ia64</code> <p>Workaround None.</p>
5.0	<p>Summary Support for multicast, database storage, and extensible security features is not provided for HP-UX 11.11.</p> <p>Workaround None.</p>
5.0	<p>Summary Setting the <code>logfile</code> parameter in the <code>tibemsd.conf</code> file to a file path containing a space causes the EMS server to create the log file in the wrong place, unless the file path is surrounded by double quotes.</p> <p>Workaround Enclose the file path in double quotation marks.</p>
5.0	<p>Summary During recovery, a server using database stores receives the following error, and startup fails:</p> <p><code>ORA-00904: "THIS_". "TXNREC_STORE_ID": invalid identifier</code></p> <p>This is related to a known issue with Hibernate.</p> <p>Workaround Restart the server. On restart, the <code>tibemsd</code> recovers correctly, with no messages lost.</p>

Identified in Release	Summary/Workaround
4.4 1-19RIVF	<p>Summary While confirming messages from a routed queue, if the daemon where queue exists is killed and restarted, some of the messages will be redelivered to the client.</p> <p>Workaround None.</p>
4.2 1-5DU9WW	<p>Summary InstallShield problems prevent uninstalling EMS from Linux 24gl23 Itanium platform.</p> <p>Workaround Uninstall using this command line (all on one line):</p> <pre>java -cp TIBCO EMS_HOME/_uninst/uninstaller.jar run</pre>
4.1 1-22ZRNM	<p>Summary JSSE cannot read PKCS12 certificates generated by some versions of OpenSSL.</p> <p>Workaround Import the certificate into a web browser; then export the certificate to a new file with extension .p12 (not .pfx).</p>