

TIBCO Enterprise Message Service™

Release Notes

*Software Release 7.0.1
March 2013*

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 © 1997-2013 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 TIBCO Documentation	ix
How to Contact TIBCO Support	ix
Release Notes	1
New Features	2
Release 7.0	2
Changes in Functionality	4
Release 7.0.1	4
Release 7.0.0	4
Deprecated and Removed Features	5
Release 7.0	5
Migration and Compatibility	8
Release 7.0	8
Closed Issues	9
Known Issues	15

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>	<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 <code>C:\tibco</code>.</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 <code>C:\tibco</code>.</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 <code>C:\tibco\ems\7.0</code>.</p>
<i>ENV_HOME</i>	
<i>EMS_HOME</i>	
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 <i>PathName</i></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> <pre>MyCommand [optional_parameter] required_parameter</pre>
	<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> <pre>MyCommand para1 param2 param3</pre>

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 <code>param1</code> and <code>param2</code>, or the pair <code>param3</code> and <code>param4</code>.</p> <pre>MyCommand {param1 param2} {param3 param4}</pre> <p>In the next example, the command requires two parameters. The first parameter can be either <code>param1</code> or <code>param2</code> and the second can be either <code>param3</code> or <code>param4</code>:</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 <code>param1</code>. You can optionally include <code>param2</code> as the second parameter. And the last parameter is either <code>param3</code> or <code>param4</code>.</p> <pre>MyCommand param1 [param2] {param3 param4}</pre>

Connecting with TIBCO Resources

How to Join TIBCOmmunity

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

How to Access TIBCO Documentation

You can access TIBCO documentation here:

<http://docs.tibco.com>

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, 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 7.0.1.

Check the TIBCO Product Support web site at <https://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 a username, you can request one. You must have a valid maintenance or support contract to use this site.

Topics

- [New Features, page 2](#)
- [Changes in Functionality, page 4](#)
- [Deprecated and Removed Features, page 5](#)
- [Migration and Compatibility, page 8](#)
- [Closed Issues, page 9](#)
- [Known Issues, page 15](#)

New Features

This section lists features added since the last major (7.0.0) release of this product.

Release 7.0

The following are new features in this release.

- **Central Administration** The Central Administration feature, accessed through a web browser, offers multiple enhancements:
 - A web-based graphical user interface (GUI) for configuring TIBCO Enterprise Message Service servers.
 - Centralized configuration, allowing administrators to apply configuration changes across multiple TIBCO Enterprise Message Service servers from a single location.

Central Administration is supported on Windows, Linux, and Mac platforms. For details, see the *TIBCO Enterprise Message Service Central Administration* guide.

- **Improved EMS Server Performance on Multi-Core Machines** This release introduces two new parameters that allow you to fine-tune performance on multi-core machines:
 - To bind network I/O threads to specific processors, set the `processor_ids` parameter in the `tibemsd.conf`.
 - To bind a storage thread to a specific processor, set `processor_id` in the `stores.conf` file.

These parameters are supported on Linux, Solaris, and Windows platforms. For more information, see the section on Performance Tuning in the *TIBCO Enterprise Message Service User's Guide*.

- **Unshared State Failover** This release adds support for unshared state connections to the C client. Three new functions have been added to the C API:
 - `tibemsUFOConnectionFactory_Create`
 - `tibemsUFOConnectionFactory_CreateFromConnectionFactory`
 - `tibemsUFOConnectionFactory_RecoverConnection`

For more information, see the *TIBCO Enterprise Message Service C & COBOL API Reference*.

- **Address Consumers with Unacknowledged Messages** The `disconnect_non_acking_consumers` parameter enables the server to identify consumers with pending messages that exceed the `maxbytes` or `maxmsgs` limits. If the limits are reached and the consumer has not acknowledged its messages, the server discards the messages sent to the consumer and disconnects the consumer's connection. This protects the server against applications that consume messages without ever acknowledging them. See the *TIBCO Enterprise Message Service User's Guide* for more information.
- **Selector Length** The administration tool can now be used to create durable subscribers using selectors longer than 2047 characters. Previously, durable subscribers with selectors containing more than 2047 characters could only be added directly to the `durables.conf` file.
- **IBM System SSL** TIBCO Enterprise Message Service installations on IBM z/OS can now configure SSL using the IBM System SSL. For more information, see Chapter 17 of the *TIBCO Enterprise Message Service C & COBOL API Reference*.
- **Support for JDK/JRE 1.7** TIBCO Enterprise Message Service now supports the Java Developer Kit (JDK) version 1.7.
- **z/Linux Platform Support** TIBCO Enterprise Message Service is now available on the z/Linux platform. Note that the z/Linux installation does *not* support Multicast or the SmartSockets Bridge.
- **Additional Platform Support** TIBCO Enterprise Message Service is now available on the following platforms:
 - Solaris 11 (SPARC)
 - Mac OS X 10.7

Changes in Functionality

This section lists changes in functionality since the last major (7.0.0) release of this product.

Release 7.0.1

The following are changes in functionality in this release.

- **Client-Only Installation** During installation, the **Customize installation** option now allows the user to install Client Only components.
- **XA Transactions** The XA protocol has been updated to remove client statefulness and the requeuing of messages is not affected by transactions.

Release 7.0.0

The following are changes in functionality in this release.

- **JSON Configuration Files** This release of TIBCO Enterprise Message Service introduces a new configuration option for EMS servers. The EMS server configuration can be stored in a single JSON-based configuration file, `tibemsd.json`. This file holds the entire configuration of the server without the need of sub-files. This configuration option is used to enable the Central Administration feature.

To convert to JSON, see the section on *Converting Server Configuration Files to JSON* in the *TIBCO Enterprise Message Service Central Administration* guide.

- **Stack Size** Previously, the EMS server and EMS C client artificially maintained a low stack size on Linux systems. However, under certain circumstances this behavior could cause stack overflows. The server and client no longer modify the system's stack size if the default is at least 1MB.

To retain the previous behavior on Linux systems, issue the `ulimit -s` command before starting the C client or EMS server.

- **OpenSSL on IBM z/OS** TIBCO Enterprise Message Service no longer supports OpenSSL on IBM z/OS systems. To implement SSL on z/OS, use IBM System SSL. For more information, see Chapter 17 of the *TIBCO Enterprise Message Service C & COBOL API Reference*.

Deprecated and Removed 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 7.0

Deprecated Features

The following features are deprecated in this release.

- **Windows XP Platform Support** Support for the Windows XP (x86 and x64) platform has been deprecated.
- **OpenLDAP** Support for openLDAP is deprecated for all Windows platforms.
- **Configuration Parameters** The following `tibemspd.conf` configuration parameters are deprecated:
 - `tibrv_xml_import_as_string`—this parameter was used to revert the `tibemspd` to an earlier, incorrect behavior. When importing messages from Rendezvous, `tibemspd` translates XML fields to byte arrays. Releases earlier than 4.0 erroneously translated them to strings. This parameter caused `tibemspd` to revert to the earlier, incorrect behavior (strings).
 - `allow_unsupported_tx_routed_q_consumers`—this parameter was used to enable an unsupported behavior: consuming messages transactionally from a routed queue. Previous releases of TIBCO Enterprise Message Service did not prevent this behavior, although it was not supported.
 - `msg_pool_size`—this parameter was used to pre-allocate a fixed pool size for message storage. After the server exhausted the pool, the server called `malloc` when it required additional storage. In a future release, only expandable pools, as configured by `msg_pool_block_size`, will be supported.
- **Dynamic Routed Queues** Dynamic routed queues are not supported. If a routed queue with wildcard is configured in the `queues.conf` or is created through the admin tool, the server issues a warning but does not prevent the queue from being created. In a future release, the server will consider a

dynamic routed queue to be a misconfiguration and will fail to start when `startup_abort_list` includes `CONFIG_ERRORS`.

Removed Features

The following features are removed in this release.

- **Support for JDK/JRE 1.5** TIBCO Enterprise Message Service supports the Java Developer Kit (JDK) version 1.6 or 1.7.
- **Platform Support** The following platforms are no longer supported:
 - HPUX 11.11 (HP PA-RISC)
 - Red Hat Enterprise Linux 4.x (x86 and x64)
 - Novell SUSE Linux Enterprise 10 (x86 and x64)
 - Oracle Enterprise Linux 4.0 (x86 and x64)

A full list of supported platforms is given in the readme file.

- **Pre 5.0 Message Store Functionality** In software releases 4.4 and earlier, EMS store files were configured in the `tibemspd.conf` file. Most configuration parameters related to this store configuration method are now obsolete and will cause a configuration error if present in the `tibemspd.conf`. If `CONFIG_ERRORS` is present in the `startup_abort_list`, the EMS server will fail to start.

These parameters are incompatible with the multiple store feature:

- `store_crc`
- `store_minimum`
- `store_minimum_async`
- `store_minimum_sync`
- `store_truncate`

The same functionality can be achieved through store configuration in the `stores.conf` file. For more information, see the section Store Messages in Multiple Stores in the *TIBCO Enterprise Message Service User's Guide*.

- **Configuration Parameters** The following `tibemspd.conf` configuration parameters are obsolete and will cause a configuration error if present in the `tibemspd.conf`. If `CONFIG_ERRORS` is present in the `startup_abort_list`, the EMS server will fail to start. The same functionality can be achieved using the current parameters shown in the table below.

Obsolete Parameter	Current Parameter
<code>client_connection_timeout</code>	<code>server_timeout_client_connection</code>

Obsolete Parameter	Current Parameter
<code>server_connection_timeout</code>	<code>server_timeout_server_connection</code>
<code>client_heartbeat</code>	<code>client_heartbeat_server</code>
<code>server_heartbeat</code>	<code>server_heartbeat_server</code>

- TIBCO Hawk Package** The `com.tibco.tibjms.admin.hawk` package is no longer included with TIBCO Enterprise Message Service. In order to use TIBCO Hawk to monitor TIBCO Enterprise Message Service, a minimum of TIBCO Hawk version 4.9 is required. As of Hawk 4.9, the `com.tibco.tibjms.admin.hawk` package is built into the installation.
- Connection Factory API** API using connection factories to create connections, SSL connections, and XA connections are now obsolete. Note that connection factory types, with the exception of generic, are also removed.

Obsolete	Replacement
<code>tibemsTopicConnectionFactory_CreateConnection</code> <code>tibemsQueueConnectionFactory_CreateConnection</code>	<code>tibemsConnectionFactory_CreateConnection</code>
<code>tibemsConnectionFactory_CreateConnectionSSL</code> <code>tibemsTopicConnectionFactory_CreateConnectionSSL</code> <code>tibemsQueueConnectionFactory_CreateConnectionSSL</code>	Use this call sequence: 1. <code>tibemsConnectionFactory_SetServerURL</code> 2. <code>tibemsConnectionFactory_SetSSLParams</code> 3. <code>tibemsConnectionFactory_SetPkPassword</code> 4. <code>tibemsConnectionFactory_CreateConnection</code>
<code>tibemsXAConnectionFactory_CreateXAConnection</code> <code>tibemsXATopicConnectionFactory_CreateXAConnection</code> <code>tibemsXAQueueConnectionFactory_CreateXAConnection</code>	<code>tibemsConnectionFactory_CreateXAConnection</code>
<code>tibemsXAConnectionFactory_CreateXAConnectionSSL</code> <code>tibemsXATopicConnectionFactory_CreateXAConnectionSSL</code> <code>tibemsXAQueueConnectionFactory_CreateXAConnectionSSL</code>	Use this call sequence: 1. <code>tibemsConnectionFactory_SetServerURL</code> 2. <code>tibemsConnectionFactory_SetSSLParams</code> 3. <code>tibemsConnectionFactory_SetPkPassword</code> 4. <code>tibemsConnectionFactory_CreateXAConnection</code>
<code>tibemsConnectionFactory_SetType</code>	None.
<code>tibemsConnectionFactory_GetType</code>	None.

Migration and Compatibility

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

Release 7.0

Users that wish to use the Central Administration feature must convert the `tibemsd.conf` server configuration files to JSON-based files. The `tibemsconf2json` utility is provided to assist in this conversion. For more information, see Appendix A, *Converting Server Configuration Files to JSON*, in the *TIBCO Enterprise Message Service Central Administration* guide.

Closed Issues

The table in this section list issues that were closed in the named releases.

Closed in Release	Key	Summary
7.0.1	EMS-4886	Fixed a memory leak in the EMS server that occurred when closing a routed consumer that had un-acknowledged messages.
7.0.1	EMS-4881	Fixed a memory leak in the EMS server that occurred when a routed consumer recovered a session while it had un-acknowledged messages.
7.0.1	EMS-4876	When purging a topic, un-acknowledged messages held by a closed topic consumer would not be removed until after the topic consumer's session was closed. Those messages are now removed during the purge.
7.0.1	EMS-4874	Fixed a defect that could cause a crash during the EMS Server startup, if a property without the "=" sign is specified in a configuration file.
7.0.1	EMS-4759	Fixed a memory leak in the EMS server that would occur when issuing the admin command <code>show durable[s] durable-name</code> , or calling the admin API method <code>getDurables(topic name)</code> .
7.0.1	EMS-4744	Fixed an error that prevented XA transactions from being rolled back or committed if the commit was issued during a failover. Now RETRY will be returned in situations where the EMS server is still available.
7.0.1	EMS-4743	Previously, if tracing was enabled but a target was not specified, the <code>target</code> property was serialized incorrectly as <code>undefined</code> , and would cause an error in the EMS server. This has been fixed.
7.0.1	EMS-4742	Fixed a message miscount issue that can occur under specific thread timing conditions during the purging of a destination.
7.0.1	EMS-4739	Fixed an error that prevented a routed queue from being edited or deleted through Central Administration after its initial configuration and deployment.
7.0.1	EMS-4736	Fixed an issue that was sometimes preventing the routing of messages when consuming on topics that were a bridge target of a global topic.

Closed in Release	Key	Summary
7.0.1	EMS-4732	Previously, a JSON-configured EMS server would report a configuration mode of XML rather than JSON when a <code>TibjmsAdmin.ServerInfo</code> object was converted to a string representation. This has been fixed.
7.0.1	EMS-4728	Previously, the <code>JMS_TIBCO_SENDER</code> property on routed messages was incorrectly populated with the originating EMS server name. This has been fixed. This field is now always populated according to the <code>sender_name</code> and <code>sender_name_enforced</code> property specifications.
7.0.1	EMS-4727	Fixed an issue that could cause a fault tolerant standby server that uses JSON configuration files to complain that the "maximum number of listen interfaces reached" after multiple failed attempts to activate (for instance because shared files were still locked).
7.0.1	EMS-4721	Fixed an error that caused an EMS server to crash when Central Administration was used to deploy a configuration with <code>client_trace</code> enabled without the target explicitly set.
7.0.1	EMS-4720	Fixed an error in software release 7.0 that required the presence of the <code>tibcrypt.jar</code> for all EMS connections. Only SSL connections now require the <code>tibcrypt.jar</code> .
7.0.1	EMS-4717	Fixed an issue that could prevent an EMS Server running on AIX platform from being added to EMS Central Administration.
7.0.1	EMS-4708	Fixed an error that could cause a message to be redelivered after it was successfully acknowledged in an XA transaction if this occurred during an EMS server fault tolerant failover.
7.0.1	EMS-4707	After deploying a server and visiting the deployment page, hitting the back button on the web browser would cause the Lock for the EMS server to change state unexpectedly and incorrectly. Only closing and reopening the page, or reloading the browser, would reset the lock. This has been fixed.
7.0.1	EMS-4699	Previously, the <code>trace_client_host</code> property in the JSON configuration had the wrong name, so any changes to this setting would have no effect. This has been fixed.

Closed in Release	Key	Summary
7.0.1	EMS-4678	Fixed an issue that could cause the EMS server to crash when a message with priority P on a topic is administratively deleted or expires in the situation where a consumer on a matching topic has pending messages but has never received a message of priority P. For example, the server could crash under the following scenario: Consumer A receives a message of priority 4 on topic foo but does not acknowledge it. Meanwhile, Consumer B on the same foo topic receives only messages with priority 8. When the unacknowledged message expires, the server crashes.
7.0.1	EMS-4666	When connecting to an EMS standby server, creating a route would make it immediately active. That is, the standby server would immediately attempt to connect to the remote server, rather than waiting for the server to become active. This has been fixed.
7.0.1	EMS-4659	If a different XA Connection was used to issue the xaEnd and xaPrepare for a given xid, the XA prepare request could be processed first by the EMS Server, resulting in error messages such as "Error processing xa prepare request, Invalid Protocol" or "Cannot find transaction referred to in xa prepare request". This has been fixed.
7.0.1	EMS-4609	Fixed an issue that would allow topic consumers with the noLocal attribute set to true to receive messages from the same connection if producers were sending messages using a local or XA transaction.
7.0.1	EMS-4580	Fixed a memory leak that affects only a standby EMS Server when a queue is created and destroyed while the server is still in standby mode. Since an admin connection uses a temporary queue to communicate with the server, the leak is observed when the connection is closed or terminated. As a result, repeatedly opening and closing an admin connection causes the process memory usage of the standby server to grow.
7.0.0	EMS-4509	Fixed an issue that prevented EMS clients from properly handling multiple threads joining the same XA transaction concurrently.
7.0.0	EMS-4500	Fixed an issue in the C client where it did not correctly compare transaction identifiers which could cause errors in the XA end, prepare, rollback or commit actions.
7.0.0	EMS-4490	Fixed a memory leak in tibemsConnectionFactory_Print.

Closed in Release	Key	Summary
7.0.0	EMS-4430	Fixed an issue that could cause the server to crash when approximately 2 billion messages were sent to consumers of a given queue between two server restarts. The limit is for each queue and does not affect topics.
7.0.0	EMS-4385	EMS C API returned XAER_INVALID for XA commit and XA start functions when using TMNOWAIT flag. This has been fixed.
7.0.0	EMS-4382	Fixed an error that could potentially cause the EMS server to crash during the recovery of prepared XA transactions.
7.0.0	EMS-4371	Fixed an issue that would cause the server to print ERROR: Abandoning transaction record due to IO failure when messages were sent using a transaction and the transaction failed because the destination limit was reached. This would also cause some internal records to accumulate in the store file, causing it to grow without possibility of reclaiming the space (using a compact admin command) until after a server restart.
7.0.0	EMS-4370	Since EMS 6.0.0, when an inbound message was part of a transaction or bridged to a destination for which the limit was reached, the resulting trace and monitor message would lack the connection information: <i>user id@host</i> . This has been fixed.
7.0.0	EMS-4359	Fixed an issue that would cause an FT standby server, with a configuration where <code>\$sys.nonfailsafe</code> is defined as an mstore, to crash when a 4.3 (or prior) admin client disconnects from the server.
7.0.0	EMS-4350	An active route configured with a host name in the URL (IP address configurations were not affected) that was disconnected would not be able to reconnect if the host name resolution failed (for instance because the DNS Service was unreachable), even after the network issue that caused the disconnect was resolved. This has been fixed.
7.0.0	EMS-4349	Previously, the EMS 64 bit library was compiled with an old version of <code>xa.h</code> where the <code>formatID</code> , <code>gtrid_length</code> and <code>bqual_length</code> were defined as "int" rather than "long" types. This has been fixed. Note that this change requires you to recompile your client applications with type "long" for their format ID and length fields.

Closed in Release	Key	Summary
7.0.0	EMS-4320	Fixed an error that sometimes caused the EMS server to abort or hang after restarting. This occurred when sending on multiple queues in the same XA transaction, where all queues are using a different type of store, and after the server shutdown before <code>xa_commit</code> .
7.0.0	EMS-4318	Resolved an issue that would prevent Java and .NET clients from using a URL with the following syntax: <code><protocol>://<port></code> For example: <code>tcp://7222</code>
7.0.0	EMS-4312	Fixed an issue that could cause the server to report corrupted records on startup for file based stores with <code>file_crc</code> enabled and <code>file_truncate</code> disabled. Note that no valid data was corrupted (only previously freed records), so no message loss would occur. Restarting the server with <code>-forceStart</code> would allow the server to start.
7.0.0	EMS-4203	Fixed an error that could cause a small memory leak in the C client when doing a JNDI lookup of a ConnectionFactory from a tibemsd.
7.0.0	EMS-4159	Fixed a defect that caused the server to crash. This could occur when a consumer closed a destination with unacknowledged messages.
7.0.0	EMS-4153	Fixed an issue with Java clients that prevented the <code>ssl_auth_only</code> parameter from being enforced. This issue arose if the EMS client could not connect to the first EMS server in the URL list and instead connected to the second server.
7.0.0	EMS-4146	Fixed an issue that could cause the EMS server to crash when restarting. This occurred when a store contained queue messages in excess of the <code>maxRedelivery</code> limit. Additionally, even if messages in the queue were properly received and acknowledged, they could be redelivered after the restart.
7.0.0	EMS-4143	Previously, the <code>link</code> option in the Makefile for HPUX did not contain <code>-lxnet</code> . This has been fixed.
7.0.0	EMS-4140	Fixed a defect that prevented the server from starting. This occurred when a listen specification used address "ANY" and used the same listen port for the primary and secondary server.
7.0.0	EMS-4117	An issue would prevent the parameter <code>dbstore_driver_password</code> from containing a clear text password that was fewer than 5 characters long. This has been fixed.

Closed in Release	Key	Summary
7.0.0	EMS-4103	Fixed an issue that could cause messages to remain in the system monitoring topic even after having been consumed by the subscriber. This occurred only when the consumer subscribed to the monitoring destination using a wildcard.
7.0.0	EMS-4063	Fixed a display error that prevented the count of dynamic queues from being reduced after a dynamic queue was promoted to a static queue.
7.0.0	EMS-4057	With the C client, when doing lookups with the same lookup context object, temporary queues used by the client to get the reply would accumulate until the lookup context was destroyed. This has been fixed.
7.0.0	EMS-4015	Fixed an error that could cause a large increase in server memory usage when purging a destination with lots of pending messages. Note that memory usage could still grow when purging the system undelivered queue (<code>\$sys.undelivered</code>).
7.0.0	EMS-3958	Fixed an error where routed servers having the same routed queue name referencing each other could cause unlimited consumers. For example, server-A has <code>queue1@server-B</code> and server-B has <code>queue1@server-A</code> . Note that this is a misconfiguration.
7.0.0	EMS-3131	Previously, the <code>EncodedProperties</code> column of table <code>EMS_MESSAGES_TABLE</code> was limited to 1024 characters. This was insufficient for database stores. TIBCO Enterprise Message Service now supports encoded properties that are greater than the column length of the <code>EncodedProperties</code> .
7.0.0	EMS-2755	When permissions were granted or revoked, the monitor message produced on <code>\$sys.monitor.admin.change</code> had the content of the two fields <code>target_dest_name</code> and <code>target_dest_type</code> permuted. That is, the field <code>target_dest_name</code> contained the type of the destination ('topic' or 'queue') and the field <code>target_dest_type</code> contained the name of that destination. This has been fixed.

Known Issues

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

Key	Summary/Workaround
7.0.0	<p>Summary On Mac OS X 10.6, the EMS server is not able to load the embedded JVM used by the Extensible Authentication and Database based store features.</p> <p>Workaround Upgrade to Mac OS X 10.7.</p>
6.3.0	<p>Summary When configuring SSL on z/Linux, authentication fails if a PEM format certificate with P7 suffix is specified to the IBM JRE.</p> <p>Workaround Modify the certificate to be in DER format with a P7 suffix.</p>
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/linux24g123/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>

Key	Summary/Workaround
5.1.4 1-AC2L2T	<p>Summary On HPUX 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>
5.0	<p>Summary The SmartSockets bridge is not supported for the 64-bit EMS server on the <code>hpux112/ia64</code> platform.</p> <p>Workaround None.</p>
5.0	<p>Summary During recovery, a server using database stores receives the following error, and startup fails:</p> <pre data-bbox="347 557 1150 583">ORA-00904: "THIS"."TXNREC_STORE_ID": invalid identifier</pre> <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>
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 <code>.p12</code> (not <code>.pfx</code>).</p>