TIBCO Enterprise Message Service™ Release Notes

Version 10.1.0

November 2021

Contents

About this Product

TIBCO is proud to announce the latest release of TIBCO Enterprise Message Service™ software.

This release is the latest in a long history of TIBCO products that leverage the power of the Information Bus® technology to enable truly event-driven IT environments. To find out more about how TIBCO Enterprise Message Service software 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 FTL.

TIBCO EMS software is part of TIBCO® Messaging.

Product Editions

TIBCO Messaging is available in a community edition and an enterprise edition.

TIBCO Messaging - Community Edition is ideal for getting started with TIBCO Messaging, for implementing application projects (including proof of concept efforts), for testing, and for deploying applications in a production environment. Although the community license limits the number of production clients, you can easily upgrade to the enterprise edition as your use of TIBCO Messaging expands.

The community edition is available free of charge. It is a full installation of the TIBCO Messaging software, with the following limitations and exclusions:

- Users may run up to 100 application instances or 1000 web/mobile instances in a production environment.
- Users do not have access to TIBCO Support, but you can use TIBCO Community as a resource (https://community.tibco.com).
- Available on Red Hat Enterprise Linux Server, Microsoft Windows & Windows Server and Apple macOS.

TIBCO Messaging - Community Edition has the following additional limitations and exclusions:

- Excludes Fault Tolerance of the server.
- Excludes Unshared State Failover.
- Excludes Routing of messages between servers.
- Excludes JSON configuration files.
- Excludes EMS OSGi bundle.
- Excludes grid store and FTL store types.

TIBCO Messaging - Enterprise Edition is ideal for all application development projects, and for deploying and managing applications in an enterprise production environment. It includes all features presented in this documentation set, as well as access to TIBCO Support.

New Features

This section lists features added since the last major (10.0.0) release of TIBCO Enterprise Message Service.

Release 10.1

New Store Types

TIBCO Enterprise Message Service now offers two additional store types that rely on non-shared storage solutions: grid stores and FTL stores. You can configure the server to store messages, state information, and configuration information in supported versions of TIBCO ActiveSpaces or TIBCO FTL, respectively. This is supported only on Linux.

Since both ActiveSpaces and FTL natively provide distributed clustered persistence, the new store types remove the requirement for EMS to interact directly with the underlying file system. In that context, EMS no longer has to rely on meeting the four shared state support criteria for a fault-tolerant setup required by the traditional file-based store, effectively removing the need for the associated physical network attached storage (SAN, NAS, etc.) and distributed file system (NFSv4, GFS2, etc.).

Grid Stores

Grid stores are designed to achieve a minimal EMS server memory footprint and quick EMS server recovery time upon failover. When configured to use grid stores, the majority of server data is stored in an ActiveSpaces data grid and is read into memory only on-demand. This approach decouples the EMS server's memory usage and failover time from the size of its stores and lends itself to quick server start-up times.

The latency costs in communicating with ActiveSpaces can make grid stores slower than file-based stores or FTL stores. The strength of grid stores lies with their scalability and consistent recovery time regardless of store size.

FTL Stores

The behavior of FTL stores is very similar to that of file-based stores. When using FTL stores, all pending persistent message data and state information is maintained in both server memory and in an FTL server cluster. Keeping this information in memory reduces the amount of communication needed with FTL and facilitates faster message processing. As with file-based stores, this approach requires that the EMS server reads the contents of all stores into memory upon start-up or failover to achieve this.

Communicating with FTL for message processing involves more overhead than the simple file reads and writes performed by file-based stores. Performance may vary depending on the usage and environment, but in general FTL stores are likely to be slower than file-based stores for that reason. The main differentiating factor between FTL stores and file-based stores lies in their underlying storage solutions and the implications of those for fault-tolerance. When making the choice between the two store types, in most cases the primary deciding factor should be whether you have access to a shared storage mechanism or not.

TLS Enhancements

- This release now supports TLSv1.3.
- This release supports a simplified way of selecting cipher suites based the OpenSSL SECLEVEL directive. We recommend using this new form.
- The Java, C, and .NET clients now allow server certificates with wildcard hostnames.
- The Java and C clients now allow hostnames in the SAN section of the certificate. Note that the .NET client doesn't.
- The server now disables TLS client side renegotiation.

General Enhancement

The trace statement shown by the server when a client ID or a durable name is too long has been enhanced (effective in EMS 8.6.0).

Release 10.0

TIBCO Enterprise Message Service 10.0 was released only on TIBCO Cloud™ Messaging.

Changes in Functionality

This section lists changes in functionality since the last major release of TIBCO Enterprise Message Service.

Release 10.1

OpenSSL

TIBCO Enterprise Message Service 10.1.0 operates with OpenSSL version 1.1.1l.

The introduction of support for TLSv1.3 has changed the way in which cipher suite lists work for the server and C client. See the *Syntax for Cipher Suites* section in the User's Guide for details.

Where appropriate, mentions of 'SSL' have been replaced with 'TLS' throughout the product and documentation.

.NET Support

Platform Requirements for .NET Core:

*   Red Hat Enterprise Linux Server
*   Microsoft Windows
*   Microsoft Windows Server

Tool Requirements:

*   To run .NET Core programs: .NET Core 3.1 (LTS)
*   To run .NET Framework programs: .NET Framework 4.8.0 (Windows only)

Package Requirements:

*   Programs that involve LDAP JNDI lookups need to reference the `System.DirectoryServices.Protocols` package. This can be done in Visual Studio or in C# project files (`*.csproj`).
*   An example is provided in the `EMS_ROOT/samples/cs` folder.

Limitations:

*   .NET Core 3.1 does not support distributed transactions.
*   LDAP JNDI lookups are supported for .NET Core only on Windows.

General Changes

*   The server now attempts to activate instead of exiting when it cannot resolve the hostname of its FT peer.
*   A new high performance memory allocator has been added to the server.
*   The `module_path` server property now only applies to the TIBCO FTL or ActiveSpaces libraries as well as a JVM. The location of the OpenSSL and compression libraries is no longer configurable. Also, the –`module_path` tibemsadmin command line parameter has been removed.
*   The tibemsadmin, tibemsmonitor tools and sample client programs no longer automatically disable verification of the server host's certificate if trusted certificates are not provided. Command line parameters are available to explicitly disable server host certificate verification if required.

Deprecated and Removed Features

The following tables list any features that have been deprecated or removed for version 10.1 of TIBCO Enterprise Message Service.

For deprecated features, if relevant, useful alternatives to the deprecated features are listed. 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.

Deprecated Features

Affected Component	Description	Deprecated in Release
Administration Tool	This release deprecates support for the Administration Tool (tibemsadmin).	10.1.0
Server Properties	This release deprecates support for the `processor_ids` server property.	8.6.0
Store Properties	This release deprecates support for the `processor_id` store property.	8.6.0
Server Properties	This release deprecates support for the `ftl_url_secondary` server property. Supply the `ftl_url` property with a pipe-separated list of URLs of FTL servers that provide realm services instead.	8.6.0
JAAS and JACI	This release deprecates support of the JAAS and JACI features on macOS. Support of these features on macOS will be removed in a future release. This does not affect the support of the JAAS and JACI features on other platforms.	8.6.0
Flow Control	The `flow_control_only_with_active_consumer` property to revert to the pre-8.4 behavior has been deprecated and will be removed in a future release.	8.4.0
JAAS and JACI	This release deprecates the `jaas_classpath` and `jaci_classpath` parameters. Users should migrate to the new `security_classpath` parameter.	8.1.0

Removed Features

Affected Component	Description	Deprecated in Release	Removed in Release
LDAP Authentication	User authentication through LDAP can be implemented either by setting the EMS server properties starting in `ldap_` or by using the LDAP JAAS authentication modules. Support of the former feature that relies on `ldap_` properties has been removed. We recommend using the newer LDAP JAAS authentication modules instead, which remain fully supported.	8.5.0	10.1.0
LDAP JNDI Lookups in C	Support by the C client of JNDI Lookups in an LDAP server has been removed. That feature is still supported by the Java and .NET clients.	8.5.0	10.1.0
Client Libraries	Support of the static C client libraries has been removed. The dynamic C client libraries remain fully supported.	8.5.1	10.1.0
Database Stores	Support of stores of type dbstore has been removed.	N/A	10.1.0
mstores	Support of stores of type mstore has been removed.	N/A	10.1.0
File Stores	The `file_crc` file-based store property has been removed. The EMS server now always uses CRC to validate data integrity when reading file-based stores.	N/A	10.1.0
Rendezvous Transports	Support for the Rendezvous transports has been removed. Pure Java Rendezvous programs are still supported.	N/A	10.1.0
Central Administration	Support of Central Administration has been removed.	N/A	10.1.0
TLS Communication	The TLSv1.1 protocol is no longer supported.	N/A	10.1.0
64-bit Symbolic Links	Upon removing support of the 32-bit server executables and C client libraries on Linux and macOS, the `-64` suffix present in the corresponding 64-bit file names had been removed and convenience symbolic links provided (such as `tibemsd64 -> tibemsd`). These symbolic links have now been removed as well.	N/A	10.1.0

Platform Support

Platform	Status	As of Release	Notes
Apple macOS 10.14	Obsolete	10.1.0	This release supports Apple macOS 10.15 and 11.
Microsoft Windows Server 2016	Removed	10.1.0	This release supports Microsoft Windows Server 2019.

Migration and Compatibility

The following are instructions on how to migrate from a previous release to version 10.1.0 of TIBCO Enterprise Message Service.

Order of Upgrade

Upon upgrading EMS software already installed on separate machines to a newer version of EMS, it is recommended to upgrade and restart in the following order:

1. Upgrade and restart all EMS servers.
2. Upgrade and restart EMS clients.

Compatibility with TIBCO FTL and TIBCO ActiveSpaces

TIBCO Enterprise Message Service release 10.1.0 is compatible with TIBCO FTL 6.7.2 or later and TIBCO ActiveSpaces 4.7.0 or later.

Migrating from Release 8.5

FTL Transports

Release 8.6.0 of TIBCO Enterprise Message Service introduced changes of behavior for FTL transports. These changes are not backward compatible in that existing FTL transport configurations in EMS and configurations in FTL are likely to require adjustments. For details, refer to the "Interoperation with TIBCO FTL > Configuration > Destination" section in the *TIBCO Enterprise Message Service™ User's Guide* and to the FTL documentation.

Closed Issues

The table lists issues closed in the listed version of TIBCO Enterprise Message Service.

Closed in Release	Key	Summary
Issues Closed in Release 10.1.0		
10.1.0	EMS-8312	The server could crash upon sending messages recovered from disk across an FTL transport. This issue affects only version 8.6.0 of TIBCO Enterprise Message Service.
10.1.0	EMS-8121	The Java and .NET implementations of the client could leak a thread upon automatically reconnecting to the server.
10.1.0	EMS-8109	The `bin/post-install.sh` script that ships on Linux and macOS reports an error on particular platforms because it is pointing to the wrong shell.
10.1.0	EMS-8048	(Windows only) A change of behavior in particular versions of the Microsoft Windows SDK results in slower console tracing in the server. This affects the overall performance of the server, depending on your `console_trace` settings. This issue affects versions 8.6.0, 8.5.1, and 8.5.0 of TIBCO Enterprise Message Service.
10.1.0	EMS-8018	(Windows only) The mechanism used by the EMS server to time out incoming TLS connection attempts is not effective if an incoming connection fails to initiate the TLS handshake. This failure can prevent the server from processing other incoming connections until the initial connection is timed out by the TCP/IP stack. This issue affects versions 8.6.0, 8.5.1, 8.5.0, and 8.4.1 of TIBCO Enterprise Message Service.
10.1.0	EMS-8002	If the `server_timeout_server_connection` or `server_timeout_client_connection` server properties are set, the server can trace an incorrect timeout value when timing out either the TLS handshake for an incoming connection request or an outgoing connection attempt.
10.1.0 (fixed in 8.6.0)	EMS-7960	The UFO implementation of the `tibemsConnectionFactory_PrintToBuffer` C API function is incorrect.
10.1.0 (fixed in 8.6.0)	EMS-7952	If the Java or .NET UFO client libraries are explicitly set to invoke the exception listener in the event of a fault-tolerant failover, they will never attempt to perform the failover process. In the case of the C UFO client library, this instead results in a crash during the failover process.
10.1.0 (fixed in 8.6.0)	EMS-7716	(Windows only) The Uninstall button for the EMS entry in the "App & features" list in Windows Setting is grayed out.
Issues Closed in Release 10.0.0		

Closed in Release	Key	Summary
10.0.0	EMS-8137	The EMS route protocol is now optimized to reduce the number of protocol messages sent under particular circumstances when a route hub-and-spoke architecture involves a large number of spokes.
10.0.0	EMS-8058	When a Java client enables server certificate verification but does not explicitly specify trusted server certificates, the client fails to automatically load the system-level trust file. This issue affects versions 8.6.0, 8.5.1, and 8.5.0 of TIBCO Enterprise Message Service.
10.0.0	EMS-8049	If a topic is configured to both import from and export to TIBCO FTL, the server crashes when messages are imported on that topic. This issue affects only version 8.6.0 of TIBCO Enterprise Message Service.
10.0.0	EMS-8024	The server cannot load DER format certificates.
10.0.0 (fixed in 8.6.0)	EMS-7713	If a connection attempt is refused due to bad credentials, the server may incorrectly warn that the connection was timed out.
10.0.0	EMS-7331	The server could leak memory when processing incoming connection requests. This issue affects versions 8.3.0-HF06 through 8.6.0-HF01 of TIBCO Enterprise Message Service.

Known Issues

The table lists known issues in the listed version of TIBCO Enterprise Message Service

Key	Summary/Workaround
EMS-8412	**Summary**: On macOS, starting one of the EMS components may result in a system prompt stating that this component *cannot be opened because the developer cannot be verified*. This happens when the installation package was downloaded through a Web browser and consequently labeled as quarantined by the operating system. **Workaround**: This will be addressed in the next release. In the meantime, a workaround consists of removing the quarantine flag from the package before unzipping it. For example: `xattr -d com.apple.quarantine TIB_ems_10.1.0_macosx_x86_64.zip`
EMS-8405	**Summary**: A disruption in the ActiveSpaces grid used by an EMS server configured with grid stores may cause the EMS server to exit. Such a disruption includes a rolling upgrade of the ActiveSpaces infrastructure. **Workaround**: Make sure to configure fault-tolerance for a standby EMS server to take over when the active server exits.
EMS-8400	**Summary**: Exporting messages through FTL transports does not work properly at this time. Please contact TIBCO Support for the current status. **Workaround**: None.
EMS-8328	**Summary**: A disruption in the FTL server cluster used by an EMS server configured with FTL stores will cause the EMS server to exit. Such a disruption includes a rolling upgrade of the FTL infrastructure. **Workaround**: Make sure to configure fault-tolerance for a standby EMS server to take over when the active server exits, and provide the necessary time for the standby server to recover all messages on activation.
EMS-8307	**Summary**: When the size of a store of type `ftl` exceeds the configured byte limit of the corresponding store in FTL, this can result in undefined behavior, possibly including the EMS server becoming unresponsive. **Workaround**: To avoid this, set the `bytelimit` parameter of the store in FTL to a large value, or to zero (unlimited). Keep in mind that an unlimited store can exhaust system memory resources in pathological cases.
EMS-8306	**Summary**: When using stores of type `ftl`, sending messages or transactions that exceed the configured maximum message size for the corresponding store in FTL will result in undefined behavior, possibly including the EMS server exiting. **Workaround**: To avoid this, set the maximum message size on the store in FTL to a large value, or to zero (unlimited). Keep in mind that sending very large messages could disrupt the quorum of persistence services, and prevent them from reforming a quorum. For more information, see the *Maximum Message Size* section in the *TIBCO FTL Development* book.

Key	Summary/Workaround
EMS-8286	**Summary**: Deleted dynamic topics may reappear after a fault-tolerance failover in a pair of servers configured with grid stores and remain even after the first scan has been completed. Note that no message is being redelivered. **Workaround**: None.
EMS- 7993	**Summary**: Since EMS 8.6.0, the server creates a single default FTL `durable` per FTL transport when the `import_subscriber_name` transport property is not set. However, this FTL durable is reported as an FTL `subscriber` in the administration tool and through the corresponding Admin API calls. **Workaround**: None.
EMS-7694	**Summary**: Installing EMS using `zypper install` on version 15 of Novell SUSE Linux Enterprise Server on x86-64 may result in the installer signaling that the corresponding EMS packages are not signed, which is expected. **Workaround**: Since the EMS RPM packages are not signed, ignore the corresponding warning.
EMS-7189	**Summary**: On certain 7.x releases of Red Hat Enterprise Linux (and all Linux distributions that are materially equivalent) older than 7.3, an EMS C client using an FT URL with its second or further member containing either `localhost` or the hostname of the local machine may not be able to connect to the EMS server on the local machine because of an issue with the hostname resolution. **Workaround**: In place of `localhost` or the local machine's hostname, provide the C client with the local machine's IP address or its loopback IP address.
EMS-7061	**Summary**: Expired messages that have been moved to the `$sys.undelivered` queue from a topic and that are consumed from `$sys.undelivered` right before stopping the server may be redelivered when the server is restarted. **Workaround**: None.
EMS-6401	**Summary**: Starting with EMS 8.2.2, the presence of a valid CRL file that is empty of revoked certificates in the `ssl_crl_path` directory will trigger a warning. Such a warning encountered at startup time will cause the EMS server to abort if the `startup_abort_list` holds the `SSL` condition. **Workaround**: If the `startup_abort_list` holds the SSL condition, make sure that no valid CRL file that is empty of revoked certificates is placed in the `ssl_crl_path` directory.

TIBCO Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, and join the TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the TIBCO Product Documentation website, mainly in HTML and PDF formats.

The website is updated frequently and is more current than any other documentation included with the product.

TIBCO Enterprise Message Service Documentation

The following documents for this product can be found on the TIBCO Enterprise Message Service ™ product documentation page:

- *TIBCO Enterprise Message Service™ Release Notes* Read the release notes for a list of new and changed features. This document also contains lists of known issues and closed issues for this release. This document is available only in PDF format.

- *TIBCO Enterprise Message Service™ Installation* Read the relevant sections of this manual before installing this product.

- *TIBCO Enterprise Message Service ™User's Guide* Read this manual to gain an overall understanding of the product, its features, and configuration.

- *TIBCO Enterprise Message Service™ C & COBOL Reference* The C API reference is available in HTML and PDF formats.

- *TIBCO Enterprise Message Service™ Java API Reference* The Java API reference can be accessed only through the HTML documentation interface.

- *TIBCO Enterprise Message Service™ .NET API Reference* The .NET API reference can be accessed only through the HTML documentation interface.

Other TIBCO Product Documentation

You may find it useful to read the documentation for the following TIBCO products:

- TIBCO® Messaging Manager
- TIBCO FTL®
- TIBCO ActiveSpaces®
- TIBCO Rendezvous®
- TIBCO® EMS Client for z/OS (CICS)
- TIBCO® EMS Client for z/OS (MVS)
- TIBCO® EMS Client for IBM i

Third-Party Documentation

- Java™ Message Service specification, available through http://www.oracle.com/technetwork/java/jms/index.html.

- *Java™ Message Service* by Richard Monson-Haefel and David A. Chappell, O'Reilly and Associates, Sebastopol, California, 2001.

- Java™ Authentication and Authorization Service (JAAS) LoginModule Developer's Guide and Reference Guide, available through http://www.oracle.com/technetwork/java/javase/jaas/index.html.

How to Contact TIBCO Support

Get an overview of TIBCO Support. You can contact TIBCO Support in the following ways:

- For accessing the Support Knowledge Base and getting personalized content about products you are interested in, visit the TIBCO Support website.
- For creating a Support case, you must have a valid maintenance or support contract with TIBCO. You also need a user name and password to log in to TIBCO Support website. If you do not have a user name, you can request one by clicking **Register** on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the TIBCO Ideas Portal. For a free registration, go to TIBCO Community.

Legal and Third-Party Notices