TIBCO® Adapter SDK

Release Notes

Software Release 5.8
November 2011

TIBCO®
The Power of Now®

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, TIBCO Administrator, TIBCO Designer, TIBCO Enterprise, TIBCO Integration Manager, TIBCO Rendezvous, and TIBCO Repository 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 © 1998-2011 TIBCO Software Inc. ALL RIGHTS RESERVED.

TIBCO Software Inc. Confidential Information

Contents

Preface

TIBCO Adapter SDK (Software Development Kit) is a class library that facilitates adapter development. All adapters implemented using TIBCO Adapter SDK have the same external interface and consistently plug in to the overall TIBCO ActiveEnterprise product suite.

Topics

Typographical Conventions

The following typographical conventions are used in this manual.

Table 1   General Typographical Conventions

Convention	Use
ENV_NAME *TIBCO_HOME* *SDK_HOME*	TIBCO products are installed into an installation environment. A product installed into an installation environment does not access components in other installation environments. Incompatible products and multiple instances of the same product must be installed into different installation environments. An installation environment consists of the following properties: • **Name** Identifies the installation environment. This name is referenced in documentation as *ENV_NAME*. On Microsoft Windows, the name is appended to the name of Windows services created by the installer and is a component of the path to the product shortcut in the Windows Start > All Programs menu. • **Path** The folder into which the product is installed. This folder is referenced in documentation as *TIBCO_HOME*. TIBCO Adapter SDK is installed into a directory within a *TIBCO_HOME*. This directory is referenced in documentation as *SDK_HOME*. The default value of *SDK_HOME* depends on the operating system. For example, on Windows systems, the default value is `C:\tibco\adapter\SDK\`*version_number*.
`code font`	Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example: Use `MyCommand` to start the foo process.
`bold code font`	Bold code font is used in the following ways: • 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
italic font	Italic font is used in the following ways: • To indicate a document title. For example: See *TIBCO ActiveMatrix BusinessWorks Concepts*. • To introduce new terms. For example: A portal page may contain several portlets. *Portlets* are mini-applications that run in a portal. • To indicate a variable in a command or code syntax that you must replace. For example: `MyCommand` *PathName*
Key combinations	Key names separated by a plus sign indicate keys pressed simultaneously. For example: Ctrl+C. Key names separated by a comma and space indicate keys pressed one after the other. For example: Esc, Ctrl+Q.
	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.

Connecting with TIBCO Resources

How to Join TIBCOmmunity

TIBCOmmunity is an online destination for TIBCO customers, partners, and resident experts, 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 All TIBCO Documentation

After you join TIBCOmmunity, 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 username and password. If you do not have a username, you can request one.

Release Notes

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

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

Release 5.8

The following are new features in this release:

- **Added Platform Support** This release adds support for Windows 7 and Windows 64-bit platforms.

- **Added Compiler Support** This release adds support for compiler MSVC++ 8.0 SP1.

Release 5.7

The following are new features in this release:

- **Support for a New Implementation for MInstance Object** A new implementation method of the MInstance object named `MInstanceListImpl`, which consumes less memory, has been added.

- **Support for RVCM Transport in Network Sink** Support for RVCM in Network Sink has been added, which guarantees receiving all of the logging messages sent from SDK.

- **Support for JMS Selector** SDK programs can access JMS selectors to filter JMS messages.

- **Support for Client ID in TIBCO Designer** A new field named `Client ID` has been added in TIBCO Designer to allow specifying the Client ID for a JMS session during configuration.

- **Support for EMS Topic Durable Name** The Adapter SDK will read and set the value specified in the field named `Durable Name` in the designer palettes of SDK.

- **Added CorrelationID Property into Response Message** The CorrelationID property value received from request messages will be provided by the Request-Response service in its response messages.

- **Support for Disabling Hawk RVD** The Hawk RVD will not be instantiated if both the standard and the class microagents are disabled.

- **Added Compiler Support** Support for the following compilers has been added:
  — compilers xlC_r v9.0 and xlC_r v11.0 on AIX platforms

    When using the compilers listed above, add the option:
    `-qnamemangling=v5`

  — compiler MSVC++ 7.1 on Microsoft Windows platforms

  — compilers CC 5.9 and CC 5.10 on Solaris SPARC platforms

  — compilers CC 5.9 and Sun Studio 12 on Solaris x86 platforms

- **Added Platform Support** Support for the following platforms has been added:
  — Red Hat Enterprise Linux 5.x (x86-64)

  — HP-UX 11i v2 (PA-RISC)

  — HP-UX 11i v3 (PA-RISC)

  — SUSE Linux Enterprise 11 (x86 and x86-64)

  — Oracle Enterprise Linux 5.x (x86 and x86-64)

Release 5.6.2

The following is the new feature in this release:

- **Support for JRE 1.6** Added support for JRE 1.6.

Release 5.6

The following are new features in this release:

- **Added Platform Support** Support for the following platforms has been added:
  — Microsoft Windows 2008 Server (x86)

  — Red Hat Enterprise Linux 4.0, 4.5, 4.6, 4.7 (x86, 32-bit and 64-bit)

  — Red Hat Enterprise Linux 5.0, 5.1, 5.2 (x86, 32-bit)

  — IBM AIX 6.1 (Power, 32-bit and 64-bit)

- **Support for Setting the RVCMQ Backlog Size** TIBCO Adapter SDK provides the following two APIs in the `MRvSession` class both for C++ and JAVA SDK to set the RVCMQ backlog size.
  — `setTaskBacklogLimitInBytes()`

  — `setTaskBacklogLimitInMessages()`

- **Support for Setting the Client ID for a JMS Connection** TIBCO Adapter SDK provides a property in the `.tra` file to set the Client ID for a JMS connection. The property name is `tibco.jmsclientid.`*session_name.*

- **Added Compiler Support** This release of TIBCO Adapter SDK adds support for compiler xlC_r v8.0 on the supported AIX platforms.

  To use this compiler add the following option: `-qnamemangling=v5`

Release 5.5.1

The following are new features in this release:

- **Added Platform Support** Support for the following platforms has been added:
  - Microsoft Windows Vista (x86)
  - Solaris 10 (x86)

Release 5.5.0

The following are new features in this release:

- **New Platforms Support**
  - AIX 5.3 64-bit
  - HP-UX 11i v3 (Itanium)

- **New Libraries Support**
  - XercesC library version 2.7
  - maverick and repowww library version 5.5.0

- **JDK Support** JDK 1.5.0

- **Messaging Software Support** TIBCO Enterprise Message Service 4.4.x

- **SSL Support at Run-time** Provides support for external SSL files at runtime by allowing applications to reference external trusted certificates instead of using the certificate in the EAR file.
  - For TIBCO Rendezvous, use the `RuntimeRvDaemonCertificate` variable and specify the path of the trusted certificate file.
  - For TIBCO BusinessWorks and TIBCO JMS, use the `RuntimeCertificatesDirectory` variable to specify the path of the directory in which all the Trusted Certificates are stored.

For more information, refer to the *TIBCO Adapter SDK Programmer's Guide*. See also the *TIBCO Designer User's Guide* for details on how to select and configure SSL.

Release 5.4.1

The following are new features in this release:

- **Added Platform Support** HP-UX 11i (PA-RISC 64-bit) is now supported.

- **Introduced a Separate Installer for Adapter SDK** From this release of the adapter, the Adapter SDK installer is separated from the TRA installer. Future hotfixes and upgrades will be directly applied to the Adapter SDK installer.

- **Support for JRE 1.5.0** Adapter SDK now supports JRE 1.5.0.

Release 5.4.0

The following are new features in this release:

- **Installer Changes** Adapter SDK now has a separate installation package. Adapter SDK is now installed in a separate directory *TIBCO_HOME*/adapter/sdk/*version*.

Release 5.3

The following are new features in this release:

- **Added platform support** The following platforms are now supported. See the Adapter SDK readme file for compiler information.

  — Sun (SPARC 32-bit) Solaris 10

  — Sun (SPARC 64-bit) Solaris 10

  — HP (PA-RISC 64-bit) HP-UX 11.11 (JAVA API only)

  — SUSE (x86) Enterprise Server 8.1, 9.3

  — IBM AS/400 V5R2, i5/OS (JAVA API only)

- **MPublisher::unRegisterListener** This method removes a durable subscription, which is the TIBCO Enterprise Message Service equivalent of TIBCO Rendezvous Certified messaging.

  If a client must receive all messages published on a topic including the ones published while the subscriber is inactive, it uses a durable subscription. The server retains a record of the durable subscription and ensures that all messages from the topic's publishers are retained until they are either

acknowledged by the durable subscriber or have expired. The `unRegisterListener` call removes the durable subscription for the subscriber.

- **MJmsEndpointSpec Class** The `messageSelector` and `durableName` member variables can now be set on the `MJmsEndpointSpec` class. See the JAVA API documentation for details.

- **Mtree::setJmsProperty() support for non string properties** Adapter SDK now supports overloaded `setJmsProperty()` methods that accept Boolean and other JMS types. This allows you to use the `setJmsProperty()` method directly, without having to convert from JMS types to `MString` before calling the method.

  The following methods have been added to `MTree`:

  — C++

```
void setJmsProperty(const MString& rsPropertyName, const bool
rsPropertyValue);
void setJmsProperty(const MString& rsPropertyName, const char
rsPropertyValue);
void setJmsProperty(const MString& rsPropertyName, const double
rsPropertyValue);
void setJmsProperty(const MString& rsPropertyName, const float
rsPropertyValue);
void setJmsProperty(const MString& rsPropertyName, const int
rsPropertyValue);
void setJmsProperty(const MString& rsPropertyName, const long
rsPropertyValue);
void setJmsProperty(const MString& rsPropertyName, const short
rsPropertyValue);
```

  — JAVA

```
void setJmsProperty(String sPropertyName, bool sPropertyValue);
void setJmsProperty(String sPropertyName, byte sPropertyValue);
void setJmsProperty(String sPropertyName, double sPropertyValue);
void setJmsProperty(String sPropertyName, float sPropertyValue);
void setJmsProperty(String sPropertyName, int sPropertyValue);
void setJmsProperty(String sPropertyName, long sPropertyValue);
void setJmsProperty(String sPropertyName, short sPropertyValue);
```

- **Support for rotation of log files based on timestamp** Previously size-based rotation of log files was supported. Now log files can be rolled over based on size or timestamp. When both size and timestamp are specified, the threshold condition reached first triggers the rollover.

  — C++

```
MFileSink.h
MFileSink(MApp* pMApp, const MString& filename, const MString&
sinkname, unsigned long fileLength, unsigned long noOfSwapFiles,
unsigned long timeInterval, const MString& fileEncoding,
MbooleanbAppendMode= Mtrue);
```

```
Overloaded constructor
virtual void setTimeInterval(unsigned long ulTimeInterval);
void calculateNextRollovetime(long timeInterval);
MConfigurationUtilities.h
```

```
MComponent* createFileSink( MString const& rsSinkName, MString
const &rsFilename ,unsigned long ulFilelength, unsigned long
ulNoOfSwapFile ,unsigned long timeInterval,MString const &
fileEncoding, MbooleanbAppendMode = Mtrue ) throw (MException);
```

— JAVA

```
MFileSink.JAVA
MFileSink(MComponentRegistry registry, String sinkName, String
fileName,int fileSizeLimit, int numFiles, int timeInterval, String
charsetName, boolean appendData)
```

```
Overloaded constructor
   int      getTimeInterval();
   void     setTimeInterval(int timeInterval);
```

```
MConfigurationUtilities.java
MComponent createFileSink(String name, String fileName, int
fileLimit, int fileCount, int timeInterval, String fileEncoding,
boolean appendMode)
```

- **Encoding Support for tracing** This feature is currently supported only in the JAVA API.

```
MFileSink.java
MFileSink(MComponentRegistry registry, String sinkName, String
fileName,int fileSizeLimit, int numFiles, int timeInterval, String
charsetName, boolean appendData)
  public String getFileEncoding()
  public synchronized void setFileEncoding(String fileEncoding);
```

- **C++ JNDI lookup support** Only the TIBCO Enterprise Message Service product is supported as the JMS provider.

Release 5.0.6

The following are new features in this release:

- **Added platform support** RedHat AS 3.0 with the GCC 3.2.3 compiler is now supported.

- **TIBCO Hawk Microagents timeout** Each standard and class Hawk Microagent now includes a timeout field. This allows you to set the amount of time the Hawk console has to wait before returning from an HMA method invoked from an application. The timeout can also be applied to custom Hawk Microagents.

   In TIBCO Designer, for standard and custom microagent, the value is configurable under an adapter's Monitoring tab. The default value is 10000 milliseconds (10 seconds).

For a custom microagent, the value resides under:

```
<adapter>
    <hawk>
        <timeout>10000</timeout>
        .....
    </hawk>
</adapter>
```

- **JMS client side reconnect parameters** Specifies the reconnect setting for client as it attempts to reconnect to the TIBCO Enterprise Message Service server after a network disconnect.

  `tibco.jmsReconnectCount` limits the number of times that the connection object attempts to reestablish a connection to the server. When this property is absent, the default value is 4. The minimum value is 1.

  `tibco.jmsReconnectDelay` is the time (in milliseconds) between reconnection attempts. When absent, the default value is 500. The minimum value is 250.

  Properties file parameters

  ```
  ---------------------
  tibco.jmsReconnectCount
  tibco.jmsReconnectDelay
  ```

  Command line options parameters

  ```
  -----------------------------
  -system:jmsReconnectCount
  -system:jmsReconnectDelay
  ```

  C++ `MAppProperties` predefined parameter

  ```
  ------------------------------------
  JMS_RECONNECT_COUNT_PROP
  JMS_RECONNECT_DELAY_PROP
  ```

  JAVA `MAppProperties` predefined parameter

  ```
  ------------------------------------
  MAppProperties.RECONNECTCOUNT
  MAppProperties.RECONNECTDELAY
  ```

- **JMS message ID, JMS correlation ID (JAVA only)** These APIs will be no operation for the TIBCO Rendezvous transport. No configuration is required. For C++, No new API can be introduced without breaking backward compatibility.

  JAVA public interfaces

  ```
  ---------------------
  void MPublisher.setJmsCorrelationId(String correlationID);
  void MPublisher.disableJmsMessageId();
  ```

In the TIBCO Enterprise Message Service, this method in effect turns message ID off. The default behavior of JMS is to set an ID for all JMS messages. For third party JMS implementations, this method's behavior is subject to the vendor's implementation.

```
void MPublisher.setPriority(int);
```

The priority is the default provided by the JMS implementation. JMS defines a ten level priority value with 0 as the lowest priority and 9 as the highest. In addition, clients should consider priorities 0-4 as gradations of normal priority and priorities 5-9 as gradations of expedited priority.

- **Global/Client Variables (JAVA only)** Note: The C++ API can already access a client variable using `MAppProperties.get()`. A global variable (that is, the default value of a variable) is not accessible in C++ since no new API can be introduced without breaking backward compatibility.

  JAVA public interface

  ```
  -----------------
  String MApp.getClientVariable(String name) throws MException
  String[][] MApp.getClientVariables()
  String MApp.getGlobalVariable(String name) throws MException
  String[][] MApp.getGlobalVariables() throws MException
  ```

- **JMS trace sink (C++ API)** The `MJmsSink` class is new in this release. The class inherits from `MSink`. In TIBCO Designer, a network sink now accepts `MJmsSession`.

  ```
  MJmsSink(MApp* pMApp, const MString& rsSinkName ,const MString&
  rsSessionName, const MString& rsSubjectName) throw
  (MException);
  virtual const char* getClassName() const;
  virtual void write(const MString& rsMessage) throw (MException);
  virtual void write(const MString& rsMessage, const MTree&
  rmTree) throw (MException);
  virtual const MString& getSinkName() const;
  virtual const MString& getDestinationName() const;
  virtual MString getDescription() const;
  MJmsSession* getJmsSession() const;
  ```

- **JMS trace sink (JAVA API)** The `MJmsSink` class is new in this release. The class inherits from `MSink`. In TIBCO Designer, a network sink now accepts `MJmsSession`.

  The class is in com.tibco.sdk.tools

  ```
  MJmsSink(MComponentRegistry registry, String name, String
  sessionName, String destinationName) throws MException
  public void write(String msg, boolean flush) Format of the
  output: traceMessage=<message>
  public void write(String msg, MTree tree, boolean flush) The
  MTree must be a JMS tree. Format of the output:
  traceMessage=<message>;traceData=<data>
  public String getDestinationName()
  ```

```
public void setDestinationName(String destinationName) throws
MException
public MJmsSession getJmsSession();
```

- **Other New JAVA APIs** The following returns the reply destination (subject) name if it exists.

  `MDataEvent.getReplyDestinationName()`

  The following returns the complete time for RVCM.

  `void MRvSession.setCompleteTime(long time) throws MException`

- **New Advisory Message** `_SDK.ERROR.JMS.RECEIVE_FAILED`

  This advisory is sent to adapters with JMS subscribers that fail to receive messages due to a JMS internal error. This can occur if the JMS server crashes or terminates during a receive operation: `jms status = SERVER_NOT_CONNECTED`. This error advisory is an unrecoverable condition. Applications that rely on this JMS session must exit.

  However, if the JMS session has a fault tolerant connection URL specified, JMS will fail over to next server. If the JMS session has a reconnect URL specified instead, this advisory occurs only after JMS has exhausted all its reconnection attempts. Note that no error or exceptions are thrown by JMS while it is attempting to reconnect.

Release 5.0.5

The following are new features in this release:

- **Added Platform Support** HP-UX 11i v2 for Intel Itanium 2 is now supported. The C++ API supports both 32-bit and 64-bit modes. The JAVA API supports only the 32-bit mode.

  Note that TIBCO Rendezvous SSL is not supported for the C++ API using the 64-bit mode.

- **New Advisory Messages (C++ only)** The following predefined advisory messages are now available:

  — `_SDK.ERROR.HAWK.BADMSGFORMAT`

    Advisory messages generated when a TIBCO Hawk microagent receives an unknown or malformed request.

  — `_SDK.WARN.HAWK.MISSING_METHOD`

    Advisory messages generated when a TIBCO Hawk method implementation is missing from the Hawk microagent for an incoming request.

Release 5.0.4

The following are new features in this release:

- **TIBCO Rendezvous SSL Support** TIBCO Adapter SDK now supports specifying secure transports in configuration for TIBCO Rendezvous sessions. This leverages the new Secure Daemon feature (using SSL) available in TIBCO Rendezvous 7.0 and later.

  For more information refer to Chapter 6 in the *TIBCO Rendezvous Concepts Guide*. See also the *TIBCO Designer User's Guide* for details on how to select and configure SSL for TIBCO Rendezvous sessions.

- **Support for selected non-TIBCO JMS Providers (JAVA only)** The JAVA Adapter SDK now allows users to specify JMS providers other than TIBCO Enterprise for JMS.

  For explicit message confirmation, the JMS standard does not support out-of-order explicit confirmation. In other words, confirming a later message implicitly confirms all earlier unconfirmed messages. This is different than how both the TIBCO Rendezvous and TIBCO Enterprise for JMS can handle explicit message confirmation. Since other providers do not provide the same degree of functionality, the adapter writer must take this behavior into account when using non-TIBCO JMS providers. Additionally, the use of SSL with non-TIBCO JMS providers is not supported in this release.

  Refer to *TIBCO Designer Palette Reference* for details on how to select and configure non-TIBCO JMS providers for your adapter.

- **Xerces library** Both C++ and JAVA are dependent on the Xerces library, which is included with the package. The version included with the package is 2.1 for C++ and 2.3 for JAVA.

- **Example program** The `/tibco/tra/5.1/resourceKit/sdk/deployableAdapter` directory contains a ZAP adapter example program that was modified to run in a TIBCO administration domain. The directory also contains step-by-step instructions on how you would change ZAP adapter to make it compatible with TIBCO Administrator.

- **Password encryption** The TIBCO Wrapper utility now allows password encryption in the adapter's properties file using the TIBCO encryption tool. The encrypted password must be prefixed with '#!'

- **JDK 1.4.2 Support** The JAVA SDK is now obfuscated.

  Microsoft Visual Studio .NET 7.1 is now supported.

Release 5.0.1

The following are new features in this release:

- **JMS Support** SDK now supports both TIBCO Rendezvous and TIBCO Enterprise for JMS as a transport. For more information, see the *TIBCO Adapter SDK Programmer's Guide.*

- **XML Support** SDK now supports XML wire format with the corresponding XML schema documents (XSDs) for use with either TIBCO Rendezvous or TIBCO Enterprise for JMS. This allows you to get an XML representation of your data and get a corresponding XSD. Only AESchema XSD is supported (generic XSD is not supported).

  For more information, see the *TIBCO Adapter SDK Programmer's Guide.*

- **TIBCO Wrapper** A wrapper utility allows your custom adapter to be installed and also to run as a Win32 service. On other platforms, it serves as a process abstraction. Using the Wrapper, it is possible to set the PATH on a per-application basis.

  For more information, see the *TIBCO Adapter SDK Programmer's Guide.*

- **Transformation Plugin** The transformation plugin allows developers to transform inbound messages, outbound messages, or both for any adapter built with the Adapter SDK.

  For more information, see the *TIBCO Adapter SDK Programmer's Guide.*

- **MDispatcher class** The `MDispatcher` class now supports overwriting per-thread stack size.

- **Xerces C** The C++ SDK uses an updated version of Xerces C. Xerces C is now a separate, dependent shared library.

- **JDK 1.4.1** The JAVA SDK supports JDK 1.4.1.

- **EUC-JP encoding** Added support for the EUC-JP encoding for C++.

- **MTrace** Added `MTrace.isRoleActive()` to the JAVA SDK.

- **Startup State** Eliminated the choice of `none` as a startup state for components. The default is now `active`.

- **MessageFormat Property** A new `tibco.messageFormat` property is supported in Properties files. You can set the same property on the command line using `–system:messageformat` *<format>*. Legal values are `aeRvMsg`, `xmlRvMsg`, and `xmlJmsMsg`.

- **Platform and Compiler Support** Added support for the Tru64 5.1A platform.

  Added support for the Microsoft Visual Studio .NET 7.0 compiler (binary targets only).

- **Documentation** The following TIBCO Adapter SDK documentation is available:

  — *TIBCO Adapter SDK Concepts* A much smaller *TIBCO Adapter SDK Concepts* manual focuses only on conceptual information. The manual has been expanded into the *TIBCO Adapter SDK Programmer's Guide*.

  — *TIBCO Adapter SDK Programmer's Guide*

  — *TIBCO Adapter SDK Message Codes*

  — The TIBCO Adapter SDK C++ and JAVA API s are now only available from the HTML interface

- **Platform and Compiler Support** Added support for the following platforms:

  — AIX 5.1

  — Tru64 5.1

  — Windows XP

  — Linux AS 2.1

- **server.ini file** If you use a TIBCO Repository Server to manage adapter files, server configuration is now stored in a file called `tibcoadmin.tra`.

Changes in Functionality

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

Release 5.8

The following feature is deprecated in this release:

Removed support for the C++ compiler: MSVC++ 7.1 on Microsoft Windows (x86) platforms.

Release 5.7

The following are changes in functionality in this release:

- **STANDBY Status** For an adapter instance that is running with fault tolerance and is in the standby mode, SDK has added a STANDBY state.

- **Logging Tracking Information** All the tracking information for the logging levels other than ERROR are now logged, instead of only logging the tracking ID as in previous releases. The tracking details can be found in the log files.

- **Xerces Changes** TIBCO Adapter SDK no longer depends on the xercesc deprecated library `libxerces-depdom.so` or `xerces-depdom_2.dll`. The third-party library Xerces 2.8 is used on all the supported platforms, replacing the former version Xerces 2.7.

- **Dropped Compiler Support** TIBCO Adapter SDK no longer supports the following compilers:

  — compiler MSVC++ 6.0 on Microsoft Windows platforms

  — compilers CC 5.3 and Sun Forte Developer 6 Update 2 J2SE cluster on Solaris SPARC platforms

  — compilers CC 5.7 and Sun Studio 10 on Solaris x86 platforms

Release 5.6.2

The following are changes in functionality in this release:

- `MData.set(name, object)` can now be used to set new values into the `MSequence` method. In the previous versions, this has been working in the C++ SDK, but not in the JAVA SDK.

Release 5.6.0

The following are changes in functionality in this release:

- The following platforms are no longer supported:

  — Solaris 7

  — Red Hat Linux 3.0

  — AIX 5.0, 5.1

  — HP 11

- TIBCO Adapter SDK can now process time zone information in the `MDateTime` class. Time zone information is added to the return value of the `toString()` method. For example, the string '2008-01-01 16:30:30 +8:00', with the time zone information included, can now be processed.

- If the decimal part of a number is '0' and its scalar type is set to 'N.0', the number will now be accepted in the MFixed class using the C++ SDK. For example, '12.0' will be accepted if its scalar type is set to '2.0'.

Release 5.5.1

There are no changes in functionality in this release:

Release 5.5

The following are changes in functionality in this release:

The following platform are no longer supported:

- HP Tru64 Unix 5.1A
  Compaq C++ v6.3

- Solaris 7 (SPARC)

Release 5.4.1

The following are changes in functionality in this release:

To assist customers in determining if their specific Linux distribution is supported, Adapter SDK now names specific vendor release versions that are supported. The following platforms are currently supported:

- Red Hat Enterprise Linux 3.0 and 4.0 (on x86)

- SuSe Linux Enterprise 9 (on x86)

Please contact TIBCO Support for information on Adapter SDK support on Linux platforms not specifically mentioned.

Release 5.4.0

The following are changes in functionality in this release:

- As Adapter SDK is now installed in a separate directory, the following changes to the makefiles are required.

    — The include files are now located in
      <TIBCO_HOME>/adapter/sdk/*<version>*/include

    — The libraries are now located in
      <TIBCO_HOME>\adapter\sdk*<version>*\bin (for Microsoft Windows)
      and *<TIBCO_HOME>*/adapter/sdk/*<version>*/lib (for UNIX).

- The Linux support policy has changed to a kernel policy. See the Adapter SDK readme file for details. As a part of this policy the following platform is no longer supported:

    — Red Hat Linux AS 2.1

- The following platform is no longer supported:

    — IBM AIX 5.1

Release 5.3.0

The following are changes in functionality in this release:

- Adapter SDK now uses ICU (International Components for Unicode) release 3.2 (previous Adapter SDK releases used ICU 1.4). The `TIB_ICU_DATA` environment setting (*<install-path>*\tibco\tra\5.3\config\g11n) is the same as in previous releases. The g11n folder contains two files, `tibicudata.dat` and `tibicudata32.dat`. `TIB_ICU_DATA` should point to `tibicudata.dat` if you are using the `maverick50` library, and should point to `tibicudata32.dat` if you are using the new `maverick53` library.

    To specify the encoding name use the charset listed in `MEncoding.h`. To maintain backward compatibility, you should use the Adapter SDK encodings listed in `MEncoding.h` and not the common alias name.

- The previously deprecated RVTX feature is completely removed from the JAVA and C++ APIs.

Release 5.0.7

The following are changes in functionality in this release:

- The Entrust jar files have been consolidated in
  `tpcl/5.2/lib/entrust/enttolkit.jar`.

- When building 64 bit applications on HP-UX Itanium (h7_ia64), you should
  not link in the libcrypto.so/sl and libssl.so/sl shared libraries.

  Also, you should not link in the libtibjms (or libtibems) shared libraries when
  building Adapter SDK applications unless the application makes direct calls
  to the tibjms (or tibems) API.

- The aelisten and aesend utilities now support the RV XML message format.

Release 5.0.6

The following are changes in functionality in this release:

- A TIBCO Rendezvous embedded license ticket is now included. You are no
  longer required to obtain a separate TIBCO Rendezvous license ticket.

- On Unix systems, the `tibrvcpp` link library is no longer needed by TIBCO
  Adapter SDK.

- To enable the use of the TIBCO Enterprise Message Service server 4.x
  reconnection feature, TIBCO Adapter SDK requires TIBCO Enterprise
  Message Service 4.x or higher for both the client and server.

- TIBCO Rendezvous 6.9 and TIBCO Rendezvous 7.1 are no longer supported.
  You must use TIBCO Rendezvous 7.2.

- The default security vendor is now Entrust.

  For applications that require SSL, you need to make sure that the classpath
  includes the Entrust jar files. See \tibco\tpcl\5.1\lib\entrust.

  If you wish to use a different security vendor, you must change the
  TIBCO_SECURITY_VENDOR java property. You can make this change in one
  of two ways

  — from the command line: -DTIBCO_SECURITY_VENDOR=<vendor>

  — In the .tra properties file for the application:
    `java.property.TIBCO_SECURITY_VENDOR= <vendor>`

  — In addition, you must make sure that vendor's libraries are included in the
    classpath.

Release 5.0.1

The following are changes in functionality in this release:

- New Classes
  - — `MEndpointSpec`
  - — `MJmsEndpointSpec`
  - — `MRvEndpointSpec`
  - — `MDispatcher` (replaces `MRvDispatcher`)
  - — `MTransformationPlugin`
  - — `MJmsSession`
- New Methods

  `MPublisher`
  - — constructor that takes an `MEndpointSpec`
  - — `setDestinationName()`
  - — `getDestinationName()`
  - — `setReplyDestinationName()`
  - — `getReplyDestinationName()`
  - — `getSession()`
  - — `getMessageFormat()`

  `MSubscriber`
  - — constructor that takes an `MEndpointSpec`
  - — `getDestinationName()`
  - — `getSession()`
  - — `getMessageFormat()`

  `MExceptionEvent`
  - — `getDestinationName()`
  - — `getReplyDestinationName()`

Directory Structure Changes

Starting with TIBCO Adapter SDK release 5.4, the TIBCO Adapter SDK directory structure has been changed. The following table lists the new directory structure. The `makefiles` for adapters built using TIBCO Adapter SDK must be changed to reflect the new directory structure.

Files	Location upto SDK 5.3	Location SDK 5.4 onwards
SDK Header files	%TIBCO_TRA_HOME%/include/sdk	%TIBCO_HOME%/adapter/sdk/*version*/include
SDK C++ Libraries	%TIBCO_TRA_HOME%/lib	%TIBCO_HOME%/adapter/sdk/*version*/lib
SDK binaries	%TIBCO_TRA_HOME%/bin	%TIBCO_HOME%/adapter/sdk/*version*/bin
SDK documentation	%TIBCO_TRA_HOME%/doc/sdk	%TIBCO_HOME%/adapter/sdk/*version*/doc
SDK examples	%TIBCO_TRA_HOME%/examples/sdk	%TIBCO_HOME%/adapter/sdk/*version*/examples
SDK JAVA libraries	%TIBCO_TRA_HOME%/lib	%TIBCO_HOME%/adapter/sdk/*version*/lib
SDK Resource Kit	%TIBCO_TRA_HOME%/resourceKit/sdk	%TIBCO_HOME%/adapter/sdk/*version*/resourceKit
SDK dlls	%TIBCO_TRA_HOME%/bin	%TIBCO_HOME%/adapter/sdk/*version*/bin
The location of the following files remain unchanged:		
Wrapper header files	%TIBCO_TRA_HOME%/include/wrapper	
Wrapper libraries	%TIBCO_TRA_HOME%/lib	
Wrapper binaries	%TIBCO_TRA_HOME%/bin	

Deprecated Features

This section describes deprecated features and lists equivalent features that accomplish the same result, if relevant. 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 equivalent feature.

Release 5.8

No features are deprecated in this release.

Release 5.7

No features are deprecated in this release.

Release 5.6.2

No features are deprecated in this release.

Release 5.0.7 (C++)

The following features are deprecated in this release:

- **Deprecated Method** `MDataEvent::getReplySender()`

  There is no replacement method.

Release 5.0.6

The following features are deprecated in this release:

- The TIBCO Rendezvous Transactional quality of service option is no longer supported. The option is still available in adapter palettes for backward compatibility.

Release 5.0.5 (Java)

The following features are deprecated in this release:

- **Deprecated Methods**
    - MTree.getNode()
    - `MTree.getNodes()`

    There are no replacement methods.

Release 5.0.4

The following features are deprecated in this release:

- Removed support for:
    - Solaris 2.6.
    - AIX 4.3.
    - HP 11 with aCC 3.27 using -AA flag.
    - HP Tru64 4.0.
    - Linux Red Hat 7.2

Release 5.0.1

The following features are deprecated in this release:

- **Deprecated features**
    - If you use a TIBCO Repository server to manage adapter files, server configuration is now stored in a file called `tibcoadmin.tra`.
    - Removed support for the CC 4.2 compiler on Solaris 2.6.

- **Deprecated C++ classes**
    - `MNode` (all methods) Use primitive types instead.
    - `MRvDispatcher` (all methods) Use `MDispatcher` instead.

- **Deprecated C++ methods**

  — `MRvSession::getCommProtocol()` Use `MSession::getDeliveryMode()` instead.

  — `MPublisher getMRvSession()` Use `MPublisher::getSession()` instead.

  — `MComponent::getProtocol` Use `MComponent::getMessageFormat` instead.

  — `MDataEvent::getReplySender(); MDataEvent::getSender(); MDataEvent::getReplySubject(); MDataEvent::getSubject()` Use `MDataEvent::sendReply()` instead.

  — `MPublisher::setSubjectName()` Use `MPublisher::setDestinationName()` instead.

  — `MPublisher::getSubjectName()` Use `MPublisher::getDestinationName()` instead.

  — `MPublisher::setReplySubject` Use `MPublisher::setReplyDestinationName()` instead.

  — `MPublisher::getReplySubject` Use `MPublisher::getReplyDestinationName()` instead.

  — `MPublisher::getProtocol()` Use `MPublisher::getDeliveryMode()` instead.

  — `MPublisher::validateMessage()` No Replacement. Use the configuration instead.

  — `MSubscriber::validateMessage()` No Replacement. Use the configuration instead.

  — `MSubscriber::getSubjectName()` Use `MSubscriber::getDestinationName()` instead.

  — `MSubscriber::getProtocol()` Use `MSubscriber::getDeliveryMode()` instead.

  — `MSubscriber::getMRvSession()` Use `MSubscriber::getSession()` instead.

- **Deprecated Java methods**

  — `MComponent.activate();` `MComponent.deactivate()` Removed from `MComponent`, but still available on `MSubscriber`.

  — `MComponent.getProtocol()` Removed from `MComponent` but available in most subclasses as `getMessageFormat()`.

  — `MDataEvent.getGUID()` Replaced by `getTrackingInfo()`

  — `MDataEvent.getSender();` `MDataEvent.getSubject();`
  `MDataEvent.getReplySender();` `MDataEvent.getReplySubject()`

  Use `MDataEvent.sendReply()` instead.

  — `MOperationDescription.addException();`
  `MOperationDescription.addParameter()` No Replacement.

  — `MPublisher.assignMsgGUID();`
  `MPublisher.getAssignMsgGUID()` Use `MTrackingInfo` instead.

  — `MPublisher.getMRvSession()` Use `MPublisher.getSession()` instead.

  — `MPublisher.setSubjectName()` Use
  `MPublisher.setDestinationName()` instead.

  — `MPublisher.getSubjectName()` Use
  `MPublisher.getDestinationName()` instead.

  — `MPublisher.setReplySubject` Use
  `MPublisher.setReplyDestinationName()` instead.

  — `MPublisher.getReplySubject` Use
  `MPublisher.getReplyDestination()` instead.

  — `MPublisher.getProtocol()` Use `MPublisher.getDeliveryMode()`
  instead.

  — `MPublisher.validateMessage()` No Replacement.

  — `MRvSession.getProtocol()` Use `MSession.getProtocol()` instead.

  — `MRvSession.getCommProtocol()` Use `MSession.getDeliveryMode()`
  instead.

  — `MRvDispatcher (all methods)` Use `MDispatcher` instead.

  — `MSubscriber.confirm()` Use `MDataEvent.confirm()` methods instead.

  — `MSubscriber.validateMessage` No Replacement.

  — `MSubscriber.getSubjectName()` Use
  `MSubscriber.getDestinationName()` instead.

  — `MSubscriber.getProtocol()` Use `MSubscriber.getDeliveryMode()`

    instead.

— `MSubscriber.getMRvSession()` Use `MSubscriber.getSession()` instead.

- **Changed Java methods** Method signatures now take `String` instead of `GlobalName`.

  — `MAssociationDescription.setEndpointA;`
  `MAssociationDescription.setEndpointA;`
  `MClassFactory.newAssociationDescription();`
  `MClassFactory.newClassDescription();`
  `MClassFactory.newClassDescription();`
  `MClassFactory.newUnionDescription();`
  `MClassFactory.loadMetadata();`
  `MClassRegistry.hasClassDescription();`
  `MClassRegistry.getClassDescription();`
  `MDataFactory.newSequence(); MDataFactory.newUnion();`
  `MModeledClassDescription.addOperation()`

- **Deprecated Java class**

  — `MNode` (all methods) No Replacement

Compatibility and Migration

The Adapter SDK C++ API is not binary backward compatible with previous releases. Support for the previously deprecated TIBCO Rendezvous TX product has been removed. The Adapter SDK JAVA API is fully backward compatible with the exception that the Rendezvous TX product is not supported.

- Existing custom adapters that do not use Rendezvous TX can upgrade to Adapter SDK by re-compiling and linking against the required libraries (as specified in the *TIBCO Adapter SDK Programmer's Guide*).

- Existing custom adapters that do not want to upgrade (or require features in Rendezvous TX) can run using the included Adapter SDK 5.0.x compatibility library without rebuilding.

Adapter SDK (both APIs) is wireformat backward compatible with previous releases and can interoperate fully with SDK 5.0.x adapters and SDK 4.1.x adapters.

Undocumented or unsupported APIs should be avoided, as the use of such may result in compilation errors, runtime errors, or both.

TIBCO Rendezvous Message Backward Compatibility

When sending and receiving `MTree` data, TIBCO Adapter SDK 5.*x* is fully compatible with TIBCO Adapter SDK 4.*x* and 3.*x*, and TIBCO Rendezvous 7.2 or later.

However, TIBCO Adapter SDK 5.*x* does not use the new data types available in TIBCO Rendezvous 7.2 and later, such as arrays or user-defined types.

TIBCO ActiveEnterprise Wire Format Backward Compatibility

The TIBCO ActiveEnterprise wire format has been extended to support hierarchical class names as well as sending `MSequence`, `MUnion` and `MAssocList` as a first class objects.

There are no restrictions in sending messages between SDK 5.x. 4.x and SDK 3.*x*.

Closed Issues

The table in this section lists issues that were closed in the Adapter SDK releases.

Fixed in Release	Defect Number	Classification	Summary
General Closed Issues			
5.8.0	ASDK-862 1-98RI5J	Runtime	While running TIBCO Adapter SDK with EMS 5.0 or higher on Microsoft Windows, an error occurred if the `libeay32.dll` file could not be found.
5.7.0	ASDK-857 1-987P6W	Runtime	Adapter crashed when processing large and complex messages in the UNIX environment due to insufficient memory for the processing.
5.7.0	ASDK-949 1-9TDAIS	Runtime	The adapter performance was low if the adapter had multiple endpoints in one JMS Session.
5.7.0	ASDK-1004 1-AV67CZ	Runtime	The JMS destination name was truncated by SDK when creating the JMS inbox.
5.5.1	1-1IJF1K	Runtime	When using pre-registration for the Certified Message publisher, specifying more than one listener on a global variable resulted in only one listener getting pre-registered.
5.5.1	1-8UB2LJ	Runtime	(Solaris only) Using custom adapters that used JMS as the transport type and were built with 64-bit SDK libraries resulted in errors.
5.5.0	1-6O7WEW	Development	The ZAP adapter example in the resource kit used the deprecated trace method.
5.5.0	1-6D3Q6H	Runtime	When an Adapter SDK JMS queue subscriber was suspended and activated again, messages pending in the JMS queue were not delivered.

Fixed in Release	Defect Number	Classification	Summary
5.5.0	1-6TMU9D	Runtime	The space of a reply parameter (of MBinary type) was freed during a `syncInvoke` call resulting in a free memory error when the reply parameter was accessed.
5.5.0	1-6SJXK9	Runtime	For multithreaded adapters, the CPU usage was very high when a JMS connection was lost and there were no incoming requests.
5.5.0	1-6T6HUN	Runtime	On Windows XP, when publishing Japanese language data using SHIFT-JIS as the encoding value, an error was displayed.
5.5.0	1-1AT61T	Runtime	On Windows 2003 Server, the value assigned to the TIB_ICU_DATA environment variable in the adapter's .tra file was not picked up at run-time.
5.5.0	1-L4VWS	Development	SDK Installer: The TIBCO Runtime Agent SDK Suite installer did not upgrade the TIBCO Adapter SDK components if installed over an existing TIBCO Runtime Agent suite install.
5.5.0	1-73A7CS	Runtime	A connection factory lookup was always done and this broke the backward compatibility. There was no way to enable or disable the connection factory lookup. A new property `tibco.connectionfactorylookup` has been added: `tibco.connectionfactorylookup = "on"/"off"` By default, the property is set to off for backward compatibility.

Fixed in Release	Defect Number	Classification	Summary
5.5.0	1-83L787	Runtime	The wrap.exe has been recompiled on the HP-UX PA-RISC platform so that the operating system can allocate more memory to running processes.
			To make use of this fix, copy and rename the wrap file to replace one or more of following executables:
			• *<TIBCO_HOME>*/tra/*<version_num>*/bin/*
			• *<TIBCO_HOME>*/designer/*<version_num>*/bin/*
			• Product-specific executables, such as *<TIBCO_HOME>*/bw/*<version_num>*/bin/*
			• administration domain specific executables, such as
			<TIBCO_HOME>/administrator/*<version_num>*/bin/ tibcoadmin_*<DOMAIN>* or *<TIBCO_HOME>*/tra/domain/*<DOMAIN>*/hawkagent _*<DOMAIN>*
5.5.0	1-81OE3Q	Runtime	For a custom adapter when the following was done and the user name was created with a password, the custom adapter could not connect to the EMS server even if the correct password was provided in the repo configuration in TIBCO Designer.
			• JMS transport was configured
			• Authentication parameter in tibemsd.conf was enabled
			• JNDI Lookups in the repo configuration was checked
			This happened because the password was not being sent to the underlying EMS APIs correctly while creating a JMS connection.
5.5.0	1-70A5QI	Runtime	EMS 4.2 could not acknowledge a message after a consumer was closed; contrary to JMS specification. When JMS Subscriber was used, this behavior had an impact on the Suspend and Activate behavior of the SDK adapter. After activation, the Subscriber did not pick up messages from the JMS server.

Fixed in Release	Defect Number	Classification	Summary
5.5.0	1-7ZVEEF	Runtime	Using the Solaris 64-bit SDK libraries, the namespace was incorrectly specified when publishing a message in RVXML or EMS. This prevented TIBCO BusinessWorks from receiving and deserializing the message properly.
5.5.0	1-3E8XQM	Development	A tunable option to configure an interval to call the `syncLedger()` method did not exist. This defect has been fixed by providing a new configuration parameter `syncLedgerTime` for synchronization of ledger file at regular intervals.
5.5.0	1-72IPDQ	Development	The manual page for MTree.to_tibrvMsg() did not exist in SDK API documentation. This has been added.
5.5.0	1-7XM4JL	Runtime	The JMS subscriber lost the current message after deactivating or activating the subscriber.
5.5.0	1-73M0UE	Development	The MPublisher.setDestinationName manual page in the SDK JAVA API documentation contained incorrect information on performance.
5.5.0	1-1X0JWL	Development	The `MSubscriber.noAutoConfirm()` method was missing from JAVA API documentation. This has been added.
5.4.1	1-6M3NQV	Runtime	The C++ date time serialization and deserialization did not work on AIX for years greater than or equal to 2038.
5.4.1	1-6VBKPE	Development	When the `aesend` utility was run with Rendezvous Certified Messaging, it exited with the following error message: `AESDKC-0030 MSession cannot convert to M_RVCM.`
5.4.0	1-6RR3ED	Development	The SDK 5.3 Release Notes had an error. The function name should have been `MSubscriber::getDestinationName()` and not `MSubscriber::getDestination()`.

Fixed in Release	Defect Number	Classification	Summary
5.4.0	1-5XZ8MB	Development	In the 5.2.1 Release Notes and API documentation, the replacement methods for `MPublisher::getProtoCol()`, `MSubscriber::getProtocol()`, and `MRvSession::getCommProtocol()` were incorrectly documented.
5.4.0	1-6Y1AQM	Development	The C++ Adapter SDK did not support the tibjmsnaming notion with JNDIlookup support. This information was not documented in the SDK documentation.
5.3.2	1-30QD6R	Development	The API documentation now explicitly states that MModeledClassDescription access methods such as getAttributeCount(), getAttributes(), etc. do not include attributes of super class(es).
5.3.2	1-5XZ8MB	Development	The TIBCO Adapter SDK API documentation for release 5.2.1 contained incorrect information about replacement methods for `MPublisher::getProtoCol()` and `MSubscriber::getProtocol()` and `MRvSession::getCommProtocol()`.
5.3.2	1-6OEV6C	Runtime	If the TIBCO Hawk Agent failed (stopped or timed-out) when redeploying an application, both the previous and current engines would run.
5.3.2	1-6OMMHP	Runtime	The SSL Verify Host option defaulted to true and the SSL Verify Hostname option defaulted to an empty string. This caused a verification failure when these options were not defined in the repository. This defect has been fixed such that SSL Verify Host defaults to false.
5.3.2	1-6Q319E	Runtime	An incorrect version of libmaverick50 was included in TIBCO Runtime Agent 5.3.0.
5.3.2	1-6SUM2Q	Runtime	The `adcicsobjectrepomapper.jar` file was not included in the *install-path*/`tra/5.x/lib/repository_mapper` directory.

Fixed in Release	Defect Number	Classification	Summary
5.3.2	1-6SUSGX	Runtime	Requesting an InputStream using Domain.getInputStream() was blocking a database connection until the InputStream was read and closed. This defect has been fixed so that a connection is blocked only when an attempt to read the stream is made.
5.3.0	1-1FLAT5	Runtime	The ICU dat file now includes encoding for ibm-5348, which is the euro supported encoding for cp1252.
Java SDK Closed Issues			
5.8.0	ASDK-647 1-6N9J4E	Runtime	COM.TIBCO.ADAPTER.getStatus() did not update "Messages Sent or Received" correctly for a request response invocation client or server (for both reliable and certified messaging).
5.7.0	ASDK-997 1-AOGJLO	Runtime	The predefined Java SDK Hawk microagent method getStatus always returned value 0 for the Total Errors field and the New Errors field even if error messages were logged after the trace function was enabled.
5.7.0	1-APK64Z	Runtime	In Java SDK, a delay was observed in the response from an asynchronous RPCC service that was multithreaded and used JMS transport on a multiprocessor system.
5.7.0	1-AMJXHP	Runtime	In Java SDK, the requests from an asynchronous RPCC service, which used JMS transport and was either multithreaded or was sharing its session with other inbound services, might be delayed.
5.7.0	ASDK-965 1-A88DMZ	Runtime	Calling the MApp.stop() method did not release RV resources.
5.6.2	1-70QJFK	Runtime	The behavior for the MSequence method in the JAVA SDK was not as same as in the C++ SDK.
5.6.2	1-9TT4S7	Runtime	In the JAVA SDK, the getAeTypeForXsiType() method in the MXsdRegistry class failed to parse the namespace.

Fixed in Release	Defect Number	Classification	Summary
5.6.0	1-8Y9C16	Runtime	In the JAVA SDK, only two threads ran concurrently even when the adapter created more than two threads for a JMS session.
5.5.0	1-7ZRM41	Runtime	The JAVA SDK did not load locale based messages: `MMessageBundle.getMessage(messageCode, args, locale)`.
5.5.0	1-7WT0AH	Development	In the MTree class, there was only the `setJmsProperty()` method available to set JMS properties. There was no corresponding method to get the properties. This defect has been fixed. You can use `MTree::to_tibjmsMsg()` to get the JMS message and invoke `tibemsMsg_GetStringProperty()` to get the property from the JMS message.
5.5.0	1-8H1WG0	Runtime	In the JAVA SDK, when using the RPC Server service, the JMS session did not close properly.
5.5.0	1-78KCKU	Development	The `MSubscriber:getEndpointSpec()` method was available in C++ but was missing in JAVA. Also, the manual page of this method was missing from API documentation.
5.5.0	1-751AB4	Runtime	The JAVA SDK did not send or receive unsigned integer correctly.
5.5.0	1-6THURS	Runtime	The `MRpcServerOperationEventListener.java` onEvent method did not throw any exception for non-AE RRI requests.
5.4.1	1-7BUSB9	Development	The `getAdvisoryListener()` method of MApp was obfuscated in the JAVA SDK 5.4.0.
5.4.1	1-72FTCB	Runtime	Configuring a `JMS_Session` with an Advisory resulted in an error in multithreaded SDK adapters.
5.3.2	1-1OSY73	Runtime	When creating an MSequence, NULL values could be inserted at any position, without getting exceptions or errors. But on the receiver-side, the sequence did not complete and ended after the first NULL value.

Fixed in Release	Defect Number	Classification	Summary
5.3.2	1-6SJXK9	Runtime	For multithreaded adapters, high CPU use would result when a JMS connection was lost even there was no incoming requests.
5.3.0	1-60H3ZX	Runtime	For adapters, an _SDK.ERROR.RPC.BADMSGFORMAT advisory error would result for a JMS request that contained empty value elements for a UTF-8 repository.
5.3.0	1-1B8USD	Runtime	JMS properties defined in message headers could not be added to JMS messages.
5.3.0	1-1WI4F6	Runtime	When `MApp.stop()` was called inside a subscriber `onEvent()` call, the JMS message just received was not confirmed.
5.3.0	1-3Z0SJQ	Development	The `MDataRvSerializer` synchronized declaration was not required and has been removed.
5.3.0	1-5ZWS7T	Runtime	`MPublisher.sendWithSyncReply()` did not work when used with the JMS transport and the publisher had the reply subject set in the repository.

Fixed in Release	Defect Number	Classification	Summary
5.0.7	1-4SSV47	Runtime	When a request was received by an adapter, it performed its processing and placed the response on a temporary queue. However, if the requester's reply listener (for example, a TIBCO BusinessWorks engine) was not running when the response was sent, the message stayed on the temporary queue "permanently", and the temporary queue remained created in the TIBCO Enterprise Message Service server.
			This defect has been fixed by providing a user configurable option of specifying a response message timeout in the properties file (or `.tra` file). The new property is "`tibco.jmsReplyTimeout.<endpoint name>`", where `<endpoint name>` is the name of the request-response server endpoint (not the service name). The value for this property is in milliseconds. For example, the following specifies a 2 second expiration time for a reply message that sent from the "`myServer`" request response server endpoint.tibco.jmsReplyTimeout.myServer 2000
5.0.7	1-4ZKC3V	Runtime	The network sink would append exceptions to subsequent trace messages.
5.0.7	1-2574UK	Runtime	The JMS message priority was not set correctly when sending. It was set only when timeout is greater than 0.
5.0.7	1-4J0CLH	Development	The aelisten utility did not display message format correctly for JMS XML and RV XML.
5.0.6	1-2LT35K	Development	`MDataEvent.getReplyDestinationName()` has been added to the JAVA library.
5.0.6	1-3K9PIB	Runtime	When a durable subscriber was configured with a global variable in its name, it was not activated even if the component startup state was active and the default startup state was not configured to active (but was set to default `none` instead).
5.0.6	1-2HSYCD	Runtime	Client-side JMS reconnection configuration parameters were not implemented.

Fixed in Release	Defect Number	Classification	Summary
5.0.6	1-2JUU55	Runtime	The JAVA standard Hawk method `getComponents()` did not return `RpcClient` or `RpcServer` endpoints like the same method in the C++ API did.
5.0.6	1-2JUU5L	Runtime	The JAVA standard Hawk method `unRegisterListener()` did not operate on an `RpcClient` endpoint. It was looking for `MPublisher`, similar to the `preRegisterListener()` method.
5.0.6	1-Y1M59	Runtime	Operation endpoints remained active after a duplicate implementation exception was thrown.
5.0.6	1-XNSNV	Development	An `RVSession` constructor used to create a CMQ session had a `timelimit` parameter that was not used and not required when creating a DQ transport.
5.0.6	1-39D7EP	Runtime	The microsecond values in `MDateTime` were incorrectly serialized. For example, 10.40 seconds would become 10.040.
5.0.6	1-1LFYRF	Runtime	A `NullPointerException` resulted when passing `args` set to null to `MMessageBundle.String messageCode, Object[] args)` and `getMessage(String messageCode, Object[] args)`.
5.0.6	1-14ZS8	Development	The `getComponents()` parameter `protocol` was unnecessary. The standard Hawk method `getComponents()` does not have `protocol` as an input parameter.
5.0.6	1-2CCHSA	Runtime	The `getStatus()` method did not display messages sent or received correctly for certified publishers and subscribers.
5.0.6	1-2H7XCP	Runtime	Time zone information can now be ignored when parsing the `dateTime` string in XML messages.
5.0.6	1-1MEU45	Runtime	When a trailing space existed in the `configURL` in an adapter properties (.tra) file, the adapter would throw `NODE NOT FOUND EXCEPTION`, even though the configured instance was in the repository.

Fixed in Release	Defect Number	Classification	Summary
5.0.6	1-1BAVV8	Runtime	A transformation plugin returning false in a JMS topic producer `sendWithSyncReply` threw an exception instead of simply stopping message flow.
5.0.6	1-26SMT2	Runtime	When using TIBCO Rendezvous certified messaging, RPC client configured preregistered listeners were not preregistered at adapter startup.
5.0.6	1-26IPCI	Runtime	When implementing a custom adapter that used a JMS subscriber with multiple dispatchers, CPU usage by the adapter would increase rapidly.
5.0.6	1-270J35	Runtime	When an adapter instance was configured with a JMS durable topic subscriber, then deployed twice, the second deployment would fail. Because both instances used the same JMS durable topic subscriber service, both used the same durable name. Since the JMS server does not allow two durable subscribers to share one durable name, the second subscriber could not register itself to the JMS server as a durable subscriber. You can now specify a global variable as part of a JMS subscriber name. For example "MyDurableSub%%InstanceId%%" will create a deployment instance specific durable name.
5.0.6	1-13KSFU	Runtime	The MMessageBundle.getMessage() method did not correctly handle the English locale resource bundle.
5.0.6	1-1WI4EH	Runtime	The JMS Queue setDestinationName() method was not working.
5.0.6	1-1WI4ET	Runtime	Durable JMS messages were not confirmed when an exception occurred.
5.0.6	1-1XMDI5	Runtime	The JAVA method, getXSD() failed for the following methods: MModeledClassDescription.getXSD() MUnionDescription.getXSD() MSequenceDescription.getXSD().
5.0.5	1-18V0NX	Runtime	A message with carriage returns sent on the JMS transport was not converted correctly on the receiving side.

Fixed in Release	Defect Number	Classification	Summary
5.0.5	1-1KMUAX	Runtime	The trace method for single variable substitution was replacing data rather substituting the variable with the data passed in.
5.0.5	1-18V0NX	Runtime	TIBCO Adapter SDK publishers sending messages over the JMS transport now preserve carriage returns in message data. `\r` is replaced with the `` escape sequence on the sending side. The XML Parser converts it to `\r` on the receiving side.
5.0.5	1-1E9NJ5, 1-1JX17P, SR:1-1E9NJ5, 1-1JX17P	Runtime	TIBCO Adapter SDK now escapes XML disallowed characters with `<? tibco-char hex-val?>`.
5.0.5	1-1IX0RH, 1-1G5JG8	Development	All destinations in TIBCO Rendezvous and in JMS are identified with an "RV subject" or a "JMS topic." `getReplySubject()` has been deprecated and replaced by `getReplyDestinationName()`. Currently `getReplySubject()` calls `getReplyDestinationName()`. The `getReplyDestinationName()` and `getDestinationName()` methods on a `MJmsProducer` now return the JMS topic or queue name.
5.0.4	1-W5I51	Runtime	Fixed a defect where multi threaded AE Operation requests were forced into serial processing instead of multi threaded/concurrent event processing.
5.0.4	1-12LP61	Runtime	A dedicated thread has been added to the Adapter SDK to handle TIBCO Hawk heartbeat events. This fixed an issue where Hawk unregistered a microagent(s) when a slow heartbeat response occurred.
5.0.4	1-M4K8H	Runtime	Fixed a defect where the transformation plugin was not being called for AE Operation client `asyncInvoke()` method calls.

Fixed in Release	Defect Number	Classification	Summary
5.0.4	1-YWF1F	Runtime	Fixed a defect where using JMS RPC Clients in a multi threaded adapter that has no other JMS endpoint types, using `asyncInvoke()`, and receiving a timeout on the request can cause the application to use 100% CPU until an event is received on RV transport, or a receiving endpoint is created on the JMS transport.
5.0.4	1-3XAQ9, 1-TNOTM	Runtime	Fixed a defect where the `getAdapterServicesInformation` Hawk method mismatches C++ and returns incorrect information.
5.0.4	1-N7X21	Runtime	Fixed a defect where the `aeXml` wireformat incorrectly transmitted certain Japanese characters.
5.0.4	1-YY9BF	Runtime	Fixed a defect where Base64 encoding operations using the included Xerces jars would fail with a java thrown `NoSuchMethodError`.
5.0.4	1-ZCE3H	Runtime	Fixed a defect where JMS sessions were not being closed properly, resulting in a resource leak on the JMS server.
5.0.4	1-XL6SD	Runtime	Fixed a defect where temporary JMS destination were not being destroyed, resulting in memory growth in the application and the JMS provider.
5.0.4	1-YBOCK	Runtime	Fixed a defect where the `aeXml` wireformat mismatched the corresponding XSD when the schema involved metadata inheritance. The order of base/derived class attributes in the XML document was incorrect.
5.0.4	1-UP5VP	Runtime	Added a requested enhancement for dynamically attempting to re-create sink files if they are removed while the SDK is using them.
5.0.4	1-WVKD6	Runtime	Fixed a defect where trailing white space was truncated in attribute values set in properties files.
5.0.4		Runtime	Fixed issue with security credential for JMS SSL connection.

Fixed in Release	Defect Number	Classification	Summary
5.0.4		Runtime	Fixed issue with enabling verify host for the 5.0.1 configuration format.
5.0.1		Runtime	The JAVA SDK wasn't calling `MPlugin` methods `onInitialization()` and `onTermination()`.
5.0.1		Runtime	Replies sent by a RVCM subscriber to an RVCM publisher are no longer certified. This fixes an issue where the ledger file grows without bound for replies to RVCM inboxes.
5.0.1		Runtime	Fixed defect where an exception was thrown when sending a message to Hawk sinks during `onInitialization()` when the Hawk microagent was not running. This is now a silent failure.
5.0.1		Runtime	Fixed defect where use of the default `MTrace` object in an application that contained multiple `MApp` instances caused messages to be traced to sinks owned by different `MApps`.
5.0.1		Runtime	Fixed a defect where `MSubscriber.getTimeout()` returned double instead of long.
5.0.1		Runtime	Fixed a defect where duplicate `loadURL` in configuration throws a duplicate class exception. Now ignores any but the first.
C++ SDK Closed Issues			
5.8.0	ASDK-700 1-717X3Z	Runtime	On Microsoft Windows, round trip serialization and deserialization did not work for date 1970-01-01.
5.8.0	ASDK-875 1-9B1WSF	Runtime	While running TIBCO Adapter SDK on Red Hat Linux 5, an error occurred if properties were created using the `MPropertiesRegistry::newSettableProperties()` method, and then are deleted using the `MPropertiesRegistry::deleteValue()` method.
5.8.0	ASDK-1773	Runtime	Memory Access Violations in XML message processing caused sporadic Illegal Operation Exceptions with SAP Adapter.

Fixed in Release	Defect Number	Classification	Summary
5.7.0	ASDK-1713	Runtime	The "Failed to convert type xxx to Unicode" or "Failed to convert Unicode to type xxx" errors occurred when encoding conversions were executed concurrently.
5.7.0	ASDK-783 1-8AHVNW	Runtime	Using the MMap to get global variables, the application crashed when the MMap was being deleted or released in the debug mode under the MSVC .net 2003 compiler.
5.7.0	1-AAASDR	Runtime	For a message using ISO-8858-1 encoding, C++ SDK failed to retrieve the value of the string type field when characters that were defined in the ISO-8858-1 character set, but not in the ASCII character set, were present in the string.
5.7.0	1-A91UQA	Runtime	In C++ SDK, when registering the Hawk microagent during the adapter start-up, the adapter crashed with a segmentation fault.
5.7.0	1-AO838S	Runtime	In the C++ SDK, the constructor of the class named MStringData did not recognize the parameter type if it was set with string variables.
5.6.2	1-6WQJFW	Runtime	In the C++ SDK, when using EMS 4.2 or higher, a TIBCO adapter failed to reconnect to the EMS server after TIBCO TRA was upgraded to a version higher than 5.3.0.
5.6.0	1-94IRV5	Runtime	In the C++ SDK, memory leaks occurred if an adapter repository with JMS SSL was created using **AE version 5.0**.
5.6.0	1-922Q9T	Runtime	In the C++ SDK, the performance level was low while deserializing a large unicode string from an AE XML format message.
5.6.0	1-9F3895	Runtime	The adapter performance level was low if JMS messages were processed in the asynchronous request response invocation service using the C++ SDK with a Multi-Core CPU on a Linux platform.

Fixed in Release	Defect Number	Classification	Summary
5.6.0	1-9F38AI	Runtime	The adapter crashed and displayed the "double free" error if JMS messages were processed in the multi-threading asynchronous request response invocation service using the C++ SDK on a Linux platform.
5.6.0	1-9CQ3KI	Runtime	While processing JMS messages with a Multi-Core CPU on Linux platforms using the C++ SDK, the adapter performance level was low.
5.6.0	1-9CF3AR	Runtime	If the decimal part of a number was '0' and its scalar type was set to 'N.0', the number was not accepted in the MFixed class using the C++ SDK. For example, '12.0' was not accepted if its scalar type was set to '2.0'.
5.5.1	1-8VVVDK	Runtime	When the timeout parameter was set to 0 (zero) and the `MOperatorRequest.asyncInvoke()` operation was invoked a memory leak was observed.
5.5.0	1-5UI0J9	Runtime	Temporary consumers were not correctly deleted in case of timeout. This caused CPU utilization to go up even during idling.
5.5.0	1-5I6PB1	Runtime	When a repoURL encrypted password contained more than 66 characters (prefixed by #!), an adapter threw the error: ".\crypto\evp\evp_enc.c(261): OpenSSL internal error, assertion failed: inl > 0".
5.5.0	1-5GVQ9N	Runtime	An adapter crashed when receiving messages from a JMS Queue where the integer type JMS property `JMSXDeliveryCount` was set.
5.5.0	1-5JB3YK	Runtime	When a C++ Adapter set the HTTP based repoURL and tried to communicate with READ-ONLY Administrator servers, an exception occurred.
5.5.0	1-7UZLT8	Runtime	In the C++ SDK, memory was utilized very rapidly when `MOperationRequest::asyncInvoke()` was used for RPC invocation in the asynchronize mode.

Fixed in Release	Defect Number	Classification	Summary
5.5.0	1-8HEES0	Development	In the C++ SDK, the data type of the "uptime" returned field of the Hawk method "getStatus" was string. This defect has been fixed.
			You can now use the "tibco.hawkUptimeDataType" property or the "-system:hawkUptimeDataType" command line parameter to specify the data type. If the value of the property or command line is "integer", the data type of the field is integer. Otherwise, the data type of the field is string.
5.5.0	1-820RT7	Runtime	The error message '"tibjmsnaming://" is not supported by EMS C API.' was unclear.
			This error message has been modified as follows:
			`AESDKC-XXXX JmsComm ERROR "tibjmsnaming://" is a TIBCO EMS JAVA API proprietary extension and is not supported by TIBCO EMS C API. Contact your TIBCO sales representative if this is your requirement.`
5.5.0	1-75HFC5	Development	The manual page for the MTimerEvent class was missing from the SDK C++ API documentation. This has been added.
5.5.0	1-7XRD9I	Development	The manual pages for to_tibrvMsg() and use_tibrvMsg() for the MTree class were missing from the SDK C++ API documentation.
5.4.1	1-7BSX6L	Runtime	On UNIX, the adapter crashed when a SDK listener configured for JMS queue was suspended and restarted.
5.4.1	1-7BSX71	Runtime	For C++ based multithreaded adapters, when the RPC Client sent a few thousand messages, the adapter crashed with a core dump. This happened in both synchronous and asynchronous scenarios of the RPC service.
5.4.1	1-6MNSY3	Runtime	The adapter crashed when a SDK listener configured for a JMS topic was suspended and restarted.
5.4.1	1-71HXIP	Runtime	When configured with JMS SSL, MQ Adapter 5.2.0 core dumped on Linux AS 3.0.

Fixed in Release	Defect Number	Classification	Summary
5.4.1	1-6T6HUN	Runtime	The value of the environment variable, `TIB_ICU_DATA`, in `Adapter.tra` file was not picked up by runtime on Windows 2003.
5.3.2	1-1Z2N4O	Runtime	When using an HTTP RepoURL, the C++ Adapter SDK was taking a long time to fetch instance configuration data from the TIBCO Administrator repository server.
5.3.2	1-5KFPC5	Runtime	ISO8859-1 is now supported as a wire encoding for the JMS transport.
5.3.2	1-6RB2TB	Runtime	The C++ Adapter SDK Base64 encoding of a binary message introduced an extra character ('\n') in the message. As a result, message decoding would fail on the JAVA subscriber side with a 'String index out of range' exception.
5.3.2	1-6TMU9D	Runtime	The space of a reply parameter (of MBinary type) was freed during a syncInvoke call, so when the reply parameter was accessed a free memory error occurred.
5.3.0	1-1B8USD	Development	JMS properties defined in message headers could not be added to JMS messages.
5.3.0	1-5MLNKR	Runtime	Specifying the JmsProviderURL with a pipe symbol resulted in an exception with an invalid port error. JMS C APIs support load balancing server URI specification only through ConnectionFactories, which TIBCO Adapter SDK code was not using. The adapter JMS session configuration should simply provide the connection factory name. The load balance url is taken care of by the JMS server, which refers to factories.conf for the definition. The JmsProviderURL on the adapter JMS session serves only to provide the initial connection url.
5.3.0	1-215ZEL	Development	`MDataEvent::sendReply` was not declared as "`const`" member function.
5.3.0	1-ICWLW	Runtime	`MInstance::setNullData` now throws `MException` if the named attribute did not exist in this instance.

Fixed in Release	Defect Number	Classification	Summary
5.3.0	1-2SCVI5	Development	`MPublisher::unRegisterListener` for JMS can now be implemented using `tibemsTopicSession_Unsubscribe()`.
5.3.0	1-3FG9TT	Runtime	On AIX, the short integer type `MInstance::toString()` did not display a correct value. -7I1 became 259I1.
5.3.0	1-5JB3YK	Runtime	When a C++ Adapter set the HTTP based repoURL and tried to communicate with READ-ONLY Administrator servers, it was throwing an exception.
5.3.0	1-3K9PIX	Runtime	If a JMS asynchronous operation invocation timed out at the same moment a reply was received, a crash could occur.
5.3.0	1-1JCF5T	Runtime	A C++ date time string before 1902 was not serializing and deserializing back to the same value.
5.3.0	1-5GVQ9N	Runtime	An adapter would crash when receiving messages from a JMS Queue where the integer type JMS property `JMSXDeliveryCount` was set.
5.3.0	1-5I6PB1	Runtime	When a repoURL encrypted password contained more than 66 characters (prefixed by #!), an adapter would throw the error: `.\crypto\evp\evp_enc.c(261): OpenSSL internal error, assertion failed: inl > 0.`
5.3.0	1-5UI0J9	Runtime	Temporary consumers were not correctly deleted in case of a timeout. This caused CPU utilization to go up even during idling.
5.3.0	1-YZ55X	Development	The `setAdvisoryListener` method would accept `MEventListener` and `MAdvisoryListener` type objects, however a `MEventListener` type object would result in an exception. The method's signature has been changed to accept only a `MAdvisoryListener` type object.
5.3.0	1-4I5DLY	Development	The `Mtree::SetProperty` method would not support non string properties (JMS primitive types).

Fixed in Release	Defect Number	Classification	Summary
5.3.0	1-5GVQ9N	Runtime	If an adapter received messages from a JMS queue where the JMS property `JMSXDeliveryCount` was set, the adapter would crash.
5.3.0	1-61K81K	Development	Calling `MTrace::trace(MString ErrorCode, MTrackingInfo*, MData*)` with no substitution string would cause an adapter to crash when the error message corresponding to the provided error code contained substitution parameter %1, %2, etc.
5.3.0	1-3O03GP	Runtime	If the value of a substitution variable was invalid for a given field, Adapter SDK reported the error as `The parameter value supplied is invalid`, but did not report the variable name.
5.3.0	1-6D3Q6H	Runtime	When an Adapter SDK JMS queue subscriber was suspended and activated again, messages pending in the JMS queue were not delivered.
5.3.0	1-6KLGK0	Runtime	When an adapter used an Adapter SDK AE operation that used the TIBCO Enterprise Message Service, no reply message was received and a time-out did not occur.
5.3.0	1-1CUCKG	Runtime	An issue with `MDataEvent::clone()` when using the JMS transport, under a rare condition, caused a crash to occur.
5.0.7	1-52ON9D	Development	The aelisten utility would core dump when listening on JMS and the JMS server was shutdown.
5.0.7	1-4TP9ZB	Runtime	The network sink trace message tag and date format were inconsistent between the C++ and JAVA API. The C++ SDK now uses the `^TRACE_MESSAGE^` field like the JAVA SDK does.
5.0.7	1-52KBEH	Runtime	If the JMS connection would fail, CPU usage would be excessively high.
5.0.7	1-4DDADX	Runtime	An invalid any type field in a JMS message could cause TIBCO Adapter SDK to crash while trying to throw an invalid input exception.

Fixed in Release	Defect Number	Classification	Summary
5.0.7	1-4SSV47	Runtime	When a request was received by an adapter, it performed its processing and placed the response on a temporary queue. However, if the requester's reply listener (for example, a TIBCO BusinessWorks engine) was not running when the response was sent, the message stayed on the temporary queue "permanently", and the temporary queue remained created in the TIBCO Enterprise Message Service server. This defect has been fixed by providing a user configurable option of specifying a response message timeout in the properties file (or .tra file). The new property is "tibco.jmsReplyTimeout.<endpoint name>", where <endpoint name> is the name of the request-response server endpoint (not the service name). The value for this property is in milliseconds. For example, the following specifies a 2 second expiration time for a reply message that sent from the "myServer" request response server endpoint. tibco.jmsReplyTimeout.myServer 2000
5.0.7	1-409HSH	Development	Fixed `MTree::setProperty()` to allow setting predefined `JMS_TIBCO` boolean properties (`JMS_TIBCO_PRESERVE_UNDELIVERED`, `JMS_TIBCO_COMPRESS` and `JMS_TIBCO_DISABLE_SENDER`) by passing the "true" or "false" string as a value.
5.0.7	1-4J0CKL	Runtime	A JMS message contained an extra `%u` used for `ae_tracking_info_`.
5.0.7	1-4SXIM3	Runtime	(TIBCO Adapter for Siebel) If the destination BC and MVL names were different, when re-opening the project, the destination MVL name was replaced with the BC name.
5.0.6	1-1OHKM1	Runtime	Special Chinese characters were missing from `TIB_ICU_DATA` and `MS950`.

Fixed in Release	Defect Number	Classification	Summary
5.0.6	1-1ZCV60	Runtime	An invocation service would crash in case of timeout when using the JMS transport if used in multi-threaded mode.
5.0.6	1-33PDG6	Runtime	`MApp::nextEvent(timelimit)` blocked longer than the time limit specified and sometimes simply hung.
5.0.6	1-2Z1JMH	Runtime	A multi-threaded RPC asynchronous client that used the Rendezvous transport would core dump.
5.0.6	1-3K9PIB	Runtime	When a durable subscriber was configured with a global variable in its name, it was not activated even if the component startup state was active and the default startup state was not configured to active (but was set to default `none` instead).
5.0.6	1-2I8CNX	Runtime	A JMS subscriber used 100% CPU when the JMS server was killed or shutdown.
5.0.6	1-2HSYCD	Runtime	Client-side JMS reconnect configuration parameters have been implemented.
5.0.6	1-2KBX1D	Runtime	Using `-system:propfile` with out an argument (or non-existing or misspelled file name) caused the adapter to core dump.
5.0.6	1-2N0KCM	Runtime	When publishing UTF-8 AE data from a publisher, `tibrvlisten` output and `rvstream` output showed the ^encoding^ as 1 instead of 2.
5.0.6	1-ADV5L	Development	`MPlugin::getProperty()` returned null. That is, it did not return the plugin property attributes specified under the plugin assoc list.
5.0.6	1-31AJOM	Runtime	In case of a `MISSING_METHOD` scenario in `MHawkMicraAgent`, the error to invoke Hawk console was not returned.
5.0.6	1-36AK31	Runtime	The file sink trace message was prefixed with a new line rather than ended with a new line.

Fixed in Release	Defect Number	Classification	Summary
5.0.6	1-1QDYV1	Runtime	The transformation plugin `stop flow` and `exception` feature did not function correctly with synchronous operation request/reply.
5.0.6	1-1OZI8Q	Runtime	The `TrackingID` generated on Windows systems produced `zzw` where the IP address should display.
5.0.6	1-1U2NDH	Runtime	High CPU utilization with `JMSQueue` Session with an inactive receiver worked only if the subscriber endpoint was activated.
5.0.6	1-1OZI7S	Runtime	`MTree.toString()` output of datetime value < 1970 and >= 2038 was incorrect.
5.0.6	1-26K13T	Runtime	Exception tracing failed for `M_THROW_NO_MEMORY` with `trackingInfo` and for a trace log, non existing error code trace.
5.0.6	1-2EX960	Runtime	The Hawk standard method `getProperties()` returned no data.
5.0.6	1-1XMDHF	Runtime	`MPlugin` now automatically appends the shared library name extension (.dll, .so, .sl) correctly.
5.0.6	1-M588D	Runtime	A meaningful error message was not posted if an incorrect property file name was specified.
5.0.6	1-1F4OAE	Runtime	If a JMS connection was stopped, an adapter terminated and did not reconnect.
5.0.6	1-TNEKG	Development	The `ReviewLedger()` method timeout value could not be configured.
5.0.6	1-1LV4TP	Development	The `aesend` and `aelisten` utilities could not use the JMS transport.
5.0.6	1-481W2	Development	`MHawkmethod` did not have a timeout parameter option.
5.0.6	1-1MEU45	Runtime	When a trailing space existed in the `configURL` in an adapter properties (.tra) file, the adapter would throw `NODE NOT FOUND EXCEPTION`, even though the configured instance was in the repository.

Fixed in Release	Defect Number	Classification	Summary
5.0.6	1-2TWYHY	Runtime	When running a multi-threaded adapter with multiple dispatchers, the JMS operation server events dispatch were not concurrent.
5.0.6	1-2ZBTJZ	Runtime	When a TIBCO BusinessWorks process sent an AE Adapter Request-Response invocation with a parameter containing empty value attribute to an adapter using UTF-8 encoding, the adapter threw a BADMSGFORMAT advisory.
5.0.6	1-270J35	Runtime	When an adapter instance was configured with a JMS durable topic subscriber, then deployed twice, the second deployment would fail. Because both instances used the same JMS durable topic subscriber service, both used the same durable name. Since the JMS server does not allow two durable subscribers to share one durable name, the second subscriber could not register itself to the JMS server as a durable subscriber. You can now specify a global variable as part of a JMS subscriber name. For example "MyDurableSub%%InstanceId%%" will create a deployment instance specific durable name.
5.0.6	1-22QT16	Runtime	Binary data to XML text (base64) encoding was not properly generated in the case where the data size was not divisible by three. This issue did not affect adapter to adapter messaging but could cause certain base64 decoders to reject it.
5.0.6	1-17UITJ	Runtime	The JMS RPC client had a memory leak.
5.0.5	1-1IPNVZ	Runtime	For the `MApp::nextEvent()` call, control to caller was not returned after timeout when no messages were in the queue
5.0.5	SR-1-1E9NJ5	Runtime	A core dump occurred when converting hex string to char during XML deserialization.

Fixed in Release	Defect Number	Classification	Summary
5.0.5	SR-1-138LJT	Development	Messages could not be touched when using TIBCO DataSecurity to encrypt and decrypt. When TIBCO DataSecurity was used in the transformation plugin, SDK touched the message which cause TIBCO DataSecurity to throw an error.
5.0.5	SR-1-1E9NJ5	Development	Escaped disallowed characters in XML.
5.0.5	1-18V0NX	Runtime	A message with carriage returns sent on the JMS transport was not converted correctly on the receiving side.
5.0.5	1-18V0NX	Runtime	TIBCO Adapter SDK publishers sending messages over the JMS transport now preserve carriage returns in message data. \r is replaced with the  escape sequence on the sending side. The XML Parser converts it to \r on the receiving side.
5.0.5	1-1E9NJ5, 1-1JX17P, SR:1-1E9NJ5 , 1-1JX17P	Runtime	TIBCO Adapter SDK now escapes XML disallowed characters with <? tibco-char hex-val?>.
5.0.5	1-1EQMKH SR 1-555FL	Runtime	TIBCO Adapter SDK now uses thread-safe versions of calls (localtime_r, gethostbyname_r, and so on) on the Solaris platform.
5.0.4	1-VGFD1	Runtime	Fixed a memory leak when using JMS transport and migrating an adapter without the suggested code changes for serialization or sending.
5.0.4	1-WWCYL	Runtime	Fixed a defect where use of the MDispatcher constructor that takes an explicit stack size throws an exception under unix platforms.
5.0.4	1-3QJ44	Runtime	Fixed a defect where MApp::setAdvisoryListener() gives an access violation when passed anything other than an MAdvisoryListener *.
5.0.4	1-4OLG1	Runtime	Fixed a defect where setting the same location for a loadURL multiple times in configuration throws an exception during startup.

Fixed in Release	Defect Number	Classification	Summary
5.0.4	1-EWSLM	Development	Fixed a defect where the `CPP_EXAMPLES.dat` repository instance file contained invalid configuration that caused TIBCO Designer to be incapable of displaying some adapter configurations correctly.
5.0.4	1-UEGGP	Development	Fixed a defect where the configure utility generates incorrect build flags for examples on HP-UX.
5.0.4	1-V8NNR	Runtime	Fixed a defect where serializing the same metadata class type across multiple threads to the `aeXml` wireformat dumps core.
5.0.4	1-Y86KD	Runtime	Fixed a memory leak for adapters using RPC Client invocations over JMS transport.
5.0.4		Runtime	Back-ported an enhancement to allow microagent name override via properties file.
5.0.4	1-XL6SD	Runtime	Fixed a defect where temporary JMS destination were not being destroyed, resulting in memory growth in the application and the JMS provider.
5.0.4	1-YBOCK	Runtime	Fixed a defect where the `aeXml` wireformat mismatched the corresponding XSD when the schema involved metadata inheritance. The order of base/derived class attributes in the XML document was incorrect.
5.0.4	1-UP5VP	Runtime	Added a requested enhancement for dynamically attempting to re-create sink files if they are removed while the SDK is using them.
5.0.4	1-WVKD6	Runtime	Fixed a defect where trailing white space was truncated in attribute values set in properties files
5.0.1		Runtime	Fixed defect on HP platforms where SIGHUP-induced failures to standard IO sinks caused all other sinks to no longer receive information.
5.0.1		Runtime	Fixed defect on Solaris where `gethostbyname()` was used instead of `gethostbyname_r()`, which was a threading problem.

Fixed in Release	Defect Number	Classification	Summary
5.0.1		Runtime	Fixed defect where hand-assembled `MTree` instances did not have tracking information attached to the message when sent.
5.0.1		Runtime	Replies sent by a RVCM subscriber to an RVCM publisher are no longer certified. This fixes an issue where the ledger file grows without bound for replies to RVCM inboxes.
Adapter Service Engine Specific Closed Issues			
5.6.1	1-9GZBEC	Runtime	Memory leaks were observed in all of the SDK services for the ActiveMatrix transport.
5.6.1	1-9OB4VT	Runtime	While invoking the SDK Request-Response Service for the ActiveMatrix transport, an exception was thrown if the adapters cloned the objects of the `MRPCRequest` class.

Known Issues

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

Defect #	Issue
ASDK-1794	**Summary** TIBCO Adapter SDK fails to start on Red Hat Enterprise Linux 6.x 64-bit platforms with an RV error. **Workaround** Backup the `rvd` executable file and then remove it from the *TIBCO_HOME*/tibrv/*version_number*/bin directory. Rename `rvd64` to `rvd` before you start TIBCO Adapter SDK.
ASDK-134 1-1AXM3B	**Summary** TIBCO Adapter SDK requires that the property `tibco.hawk.microagent.name` be set to register custom hawk methods. TIBCO Administrator sets the property `tibco.hawk.microagent.name` when an adapter is deployed. If the property `tibco.hawk.microagent.name` is not set (the adapter is not deployed and the property is not set in the adapter `.tra` file), the observed behavior is that sometimes methods appear in the Hawk Display and other times they do not. **Workaround** Deploy the adapter from TIBCO Administrator or set the property `tibco.hawk.microagent.name` in the adapter `.tra` file.