
JBoss Transaction manager 4.16

Release Notes

Release Notes for JBoss Transaction Manager



Misty Stanley-Jones

Copyright © 2011 JBoss.org.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at <http://creativecommons.org/licenses/by-sa/3.0/>. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, Fedora, the Infinity Logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux® is the registered trademark of Linus Torvalds in the United States and other countries.

Java® is a registered trademark of Oracle and/or its affiliates.

XFS® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

All other trademarks are the property of their respective owners.

Abstract

Release notes for ArjunaCore, ArjunaJTA, ArjunaJTS, and XTS components of the JBoss Transaction Manager.

1. ArjunaCore	2
2. ArjunaJTA	3
3. ArjunaJTS	3
4. XTS	3
5. Recovery	4
6. JCA	4
A. Revision History	4

1. ArjunaCore

New features

JBTM-573 ArjunaCore now includes a command-line tool for manually pruning log files.

TBD

JBTM-796 All ObjectStore types supported by the ObjectStoreAPI are now supported by MBean **ObjStoreBrowserMBean**.

Originally, **ObjStoreBrowserMBean** was only known to support file-based ObjectStores. It now implements the ObjectStoreAPI, and works with all types of ObjectStores supported by the API.

Resolved issues

JBTM-575 RollbackException now behaves as it did in JBoss Application Server 4.3 and earlier, providing debugging information about the cause of the exception.

In JBoss Application Server 4.2 and previous, when a transaction rolled back, the RollbackException included the root cause of the exception. In JBoss Application Server 5.1, the exception no longer included this information, even though the information can still be found in a different area of the log file. Code which relied on the root cause information included in the exception no longer worked as expected.

The debugging information has been put back into the exception, so that it works as it did previously. The following algorithm is used:

1. If `setRollbackOnly()` is called before `commit()`, the *initCause* is **setRollbackOnly called from...**, regardless of anything that may go wrong during the commit. `beforeCompletions` are not called.
2. If `getDeferredThrowable` is not **null**, the *initCause* is the `deferredThrowable`, even if **setRollbackOnly** is also called before the exception is received.
3. If **setRollbackOnly** was called, it is the root cause. This covers the case where a **beforeCompletion** calls `setRollbackOnly` but does not throw an exception.

This fix allows you to code against the root cause of the RollbackException.

JBTM-814 TSR implementations now implement **ObjectFactory**.

TSR implementations have been modified to implement **ObjectFactory**.

JBTM-811 The finalizer has been removed from the **StateManager**.

The **StateManager** no longer includes the finalizer.

JBTM-599 A synchronization in CacheStore has been fixed.

Removing threads operating on AsyncStore did not check to see if the cache was full, and did not notify AsyncCache to wake up. This could cause a stalling condition, if there were more removes than adds to the cache. Removes now check to see if the cache is full, and notify AsyncCache accordingly.

2. ArjunaJTA

New features

JBTM-77 **UserTransaction** is now serializable and referencable, in compliance with the JTA.

The Java Transactions API (JTA) specifies that **UserTransaction** needs to be serializable and referenceable. JBossTA now complies with this requirement.

3. ArjunaJTS

Resolved issues

JBTM-816 Threads in several components have been given names, for easier profiling.

The threads in the transaction reaper, reaper worker, action store scanner, and the XTS reaper worker have been given names. This makes them easier to identify during profiling.

4. XTS

New features

JBTM-798 XTS service and initialization classes are now packaged in a JAR, for easy deployment outside JBossAS.

XTS service and initialization classes are now packaged into a separate JAR, to make it easier to deploy XTS into other application servers, such as Tomcat.

JBTM-799 You can now programmatically configure the WS-C and WS-T service endpoint URLs and paths.

In the past, WS-C and WS-T service endpoint URLs were hard-coded to conform to the pattern **http://bindAddress:bindPort/war-name/service-name**. This made it more difficult to deploy these services in different application servers. The path portion of the URLs of the service endpoints are now configurable.

Resolved issues

JBTM-518 WST 1.1 support is now included in the default configuration

WS-BA 0.2.1 only imported the WST 1.0-compliant class libraries, and required a patch to import the WST 1.1-compliant class libraries. The BA framework now includes a **libs/** directory that hosts the WST 1.1-compliant libraries, so WST 1.0 and 1.1 are both supported in the default configuration.

JBTM-800 Deploying just the Coordinator component of XTS no longer causes an exception to be thrown.

The configuration instructions for deploying just the Coordinator component of XTS used to cause an exception, because the `ATParticipantRecoveryModule` was not available on the Participant component. The Participant and Coordinator recovery modules now create and install a manager if one is not already found, and the exception is no longer thrown.

JBTM-803 WSDLs are no longer duplicated in the WS-C and WS-TX JARs.

The WSDLs have been removed from the WS-TX JAR. Both the WS-C and WS-TX JAR are needed by XTS, and the duplicate information was causing confusion without serving a purpose.

JBTM-804 The log of an invalid state SOAP fault now correctly reports that the participant is compensated, not cancelled.

An invalid state SOAP fault causes the participant to be compensated, but the log message used to report that was cancelled. This has been fixed, so that the log message now agrees with the actual action.

JBTM-816 Threads in several components have been given names, for easier profiling.

The threads in the transaction reaper, reaper worker, action store scanner, and the XTS reaper worker have been given names. This makes them easier to identify during profiling.

5. Recovery

Resolved issues

JBTM-797 ORB setup now detects exceptions due to in-use ports.

If the ORB was unable to bind to the specified port, it would print a warning to standard output and remain in an unusable state. No exception was thrown and there was no mechanism for making the ORB try again to bind to the port. The ORB now throws an exception and the developer can make the ORB try again to bind to the port.

JBTM-816 Threads in several components have been given names, for easier profiling.

The threads in the transaction reaper, reaper worker, action store scanner, and the XTS reaper worker have been given names. This makes them easier to identify during profiling.

6. JCA

Resolved issues

JBTM-812 JDBC3 code is no longer included.

JDBC3 code was required for building on JDK 5, but this is no longer needed, and has been removed.

A. Revision History

Revision 1.0 Wed Dec 8 2010

JBossTS 4.16 Release Notes

Misty Stanley-Jones misty@redhat.com