

JBoss Transactions 4.2

Release Notes

JBTS-RN-7/18/06



Legal Notices

The information contained in this documentation is subject to change without notice.

JBoss Inc. makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. JBoss Inc. shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Java™ and J2EE is a U.S. trademark of Sun Microsystems, Inc. Microsoft® and Windows NT® are registered trademarks of Microsoft Corporation. Oracle® is a registered U.S. trademark and Oracle9™, Oracle9 Server™ Oracle9 Enterprise Edition™ are trademarks of Oracle Corporation. Unix is used here as a generic term covering all versions of the UNIX® operating system. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

Arjuna is a trademark of Hewlett-Packard Company and is used here under licence.

Copyright

JBoss, Home of Professional Open Source Copyright 2006, JBoss Inc., and individual contributors as indicated by the @authors tag. All rights reserved.

See the copyright.txt in the distribution for a full listing of individual contributors. This copyrighted material is made available to anyone wishing to use, modify, copy, or redistribute it subject to the terms and conditions of the GNU General Public License, v. 2.0. This program is distributed in the hope that it will be useful, but WITHOUT A WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

See the GNU General Public License for more details. You should have received a copy of the GNU General Public License, v. 2.0 along with this distribution; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, * MA 02110-1301, USA.

Software Version

JBoss Transactions 4.2.

Restricted Rights Legend

Use, duplication, or disclosure is subject to restrictions as set forth in contract subdivision (c)(1)(ii) of the Rights in Technical Data and Computer Software clause 52.227-FAR14.

© Copyright 2006 JBoss Inc.



Contents

About This Guide 5

What This Guide Contains..... 5

Audience 5

Prerequisites..... 5

Organization..... 5

Documentation Conventions 5

Additional Documentation..... 6

Contacting Us 6

Release notes 7

Enhancements 7

Fixed issues..... 7

Known issues 7

About This Guide

What This Guide Contains

The Release Notes contains important information on changes to JBoss Transactions 4.2 since the last release and information on any outstanding issues.

Audience

This guide is most relevant to engineers who are responsible for administering JBoss Transactions 4.2 installations.

Prerequisites

None.

Organization

This guide contains the following chapters:

- **Chapter 1, Release notes:** contains the actual release notes.

Documentation Conventions

The following conventions are used in this guide:

| Convention | Description |
|----------------------------|--|
| <i>Italic</i> | In paragraph text, italic identifies the titles of documents that are being referenced. When used in conjunction with the Code text described below, italics identify a variable that should be replaced by the user with an actual value. |
| Bold | Emphasizes items of particular importance. |
| Code | Text that represents programming code. |
| Function Function | A path to a function or dialog box within an interface. For example, "Select File Open." indicates that you should select the Open function from the File menu. |
| () and | Parentheses enclose optional items in command syntax. The |

| | |
|-----------------|---|
| | vertical bar separates syntax items in a list of choices. For example, any of the following three items can be entered in this syntax: <code>persistPolicy (Never OnTimer OnUpdate NoMoreOftenThan)</code> |
| Note: | A note highlights important supplemental information. |
| Caution: | A caution highlights procedures or information that is necessary to avoid damage to equipment, damage to software, loss of data, or invalid test results. |

Table 1 Formatting Conventions

Additional Documentation

In addition to this guide, the following guides are available in the JBoss Transactions 4.2 documentation set:

- JBoss Transactions 4.2 *Administration Guide*: Provides information on administering JBoss Transactions 4.2.
- JBoss Transactions 4.2 *Installation Guide*: This guide provides instructions for installing JBoss Transactions 4.2.
- JBoss Transactions 4.2 *Programmer's Guide*: Provides guidance for writing applications.
- JBoss Transactions 4.2 *Quick Start Guide*: Getting started with JBossTS; not for a novice user.
- JBoss Transactions API *Programmer's Guide*: Provides guidance when using the JTA for building transactional applications.
- JBoss Transactions *Failure Recovery Guide*: Describes the failure recovery aspects of JBossTS.
- TxCore *Programmer's Guide*: Describes how to write transactional applications using the non-distributed transaction engine at the heart of JBossTS.

Contacting Us

Questions or comments about JBoss Transactions 4.2 should be directed to our support team.

Release notes

Enhancements

This is the first release of JBoss Transactions, formerly the Arjuna Transaction Service. In order to facilitate migration of existing ArjunaTS users, there have been no changes to the structure or deployment in this release of JBossTS. However, in subsequent releases there will be modifications to the package structure and other updates, so please continue to check the Release Notes.

The following enhancements have been made to the 4.2 GA version of *JBossTS*:

- This release supports the Last Resource Commit Optimization for resources that are not two phase aware. This functionality is only supported by the JTS Transaction Manager.
- This release supports JBoss Application Server 4.0.3SP1.

Fixed issues

The following issues have been identified and fixed:

- During recovery, JBossTS would report problems with badly formatted UIDs. This was identified as <http://jira.jboss.com/jira/browse/JBTM-94> and has now been fixed.
- If a non-zero transaction timeout was specified, then JBossTS did not correctly propagate the timeout value to XAResource instances. This was identified as <http://jira.jboss.com/jira/browse/JBTM-93> and has now been fixed. Furthermore the setting of the timeout on the XAResource is now configurable. See the JTA Programmers Guide for more details.

Known issues

Issues that are known in this release are:

1. When running within the JBoss Application Server, there has been no attempt at modifications of either JBossMQ or JBossMessaging to make use of JBoss Transactions.
2. The Web Services transactions component is not available in the 4.2 release, but will be available in a 4.2.1 release.
3. The Last Resource Commit Optimization is not supported by the JTA Transaction Manager.

4. The Last Resource Commit Optimization is not supported by the JTS Transaction Manager when interposition is being used. The value of the property `com.arjuna.ats.jts.contextPropMode` must be set to `CONTEXT`.
5. There is an issue with Oracle 9.2 (and some older versions of Oracle 8) to do with transaction serializability. It is not possible to set the transaction isolation level to serializable on JDBC connections used in global (XA) transactions. Normally transaction isolation levels are set on a connection obtained from an `XAConnection`. However, if the isolation level is changed from the default (read committed) to serializable, Oracle throws an `XAException` (`XAER_RMERR` and Oracle `XAException` code 24776) whenever `XAResource.start` is called (even if only on a single resource). As such, if you want to use Oracle you can only do so with the default isolation level. If you are using the Arjuna JDBC driver, this can be obtained by setting the property `com.arjuna.ats.jdbc.isolationLevel` to `TRANSACTION_READ_COMMITTED`. We are investigating other solutions to this issue. (Note, see `misc/oraclexa.java` for example code that illustrates the issue.)
6. The current OTS contains corrections to several errors in the `CosTransactions idl` which have yet to be fixed. Although these errors do not cause problems in several of the leading commercial ORBs we have had to impose our own corrections to the `CosTransactions idl` in order to get *JBossTS* to run on other, more OMG compliant ORBs. It is unlikely that the modifications we have made will correspond exactly to the final changes the OMG will make. The `CosTransactions.idl` and `ArjunaOTS.idl` files used in this document are idealised versions, i.e., how they should be if all ORBs worked correctly. You should always examine the idl files installed with your version of *JBossTS* to determine any differences.
7. In previous drafts of the OTS, there were several name clashes between interface names and variable names. These include, the variable names `coordinator` and `terminator` in the `TransIdentity` structure. These have been changed to `coord` and `term`. However, the variable `current` (used in the same structure), which clashes with the `Current` interface, has not been changed; we have changed this to `currentTransaction`.
8. Several ORBs cannot cope with anonymous sequences, i.e., sequence `<octet> tid`. Therefore, several `typedefs` have been added to the idl specifically for these ORBs.
9. The JacORB POA has a minimum and maximum number of threads available to it in it's thread pool to process invocations. These values are set in the `jacorb.properties` file as `jacorb.poa.thread_pool_max` and `jacorb.poa.thread_pool_min`. If these variables are not tailored to your application the POA can get into a state where it cannot process requests because the pool has been exhausted and the 'in-use' request processors are waiting to make invocations on the same POA. This has the same apparent affect as your application being deadlocked.
10. The trail map accompanying *JBossTS* does not work correctly if it is installed in a directory containing spaces.
11. When doing explicit interposition on JDBC/XA connections, the calling order should be:

```
begin()
registerTransaction()
getConnection()
```

close()
unregisterTransaction()
commit() / rollback()

If the close() is done after the unregisterTransaction(), Oracle will throw a oracle.jdbc.xa.OracleXAException() when the next transaction is started.