JBoss Enterprise Application Platform 5

Release Notes

for Use with JBoss Enterprise Application Platform 5

Edition 5.0.0

Last Updated: 2017-10-18

JBoss Enterprise Application Platform 5 Release Notes

for Use with JBoss Enterprise Application Platform 5
Edition 5.0.0

Isaac Rooskov
irooskov@redhat.com

Laura Bailey

Legal Notice

Abstract

These release notes contain important information related to JBoss Enterprise Application Platform 5.0.0 that may not be currently available in the product manuals. You should read these Release Notes in their entirety before installing JBoss Enterprise Application Platform 5.0.0.

Table of Contents

1. INTRODUCTION

These release notes contain important information related to JBoss Enterprise Application Platform 5.0.0. New features, known problems, resources, and other current issues are addressed here.

1.1. Overview

JBoss Enterprise Application Platform is the next evolutionary step in open source enterprise software. It is a powerful tool for developing rich, high performance, Web 2.0 applications on a pure Java Platform.

Further, by integrating best-of-breed open source frameworks such as JBoss Seam, Hibernate, Tomcat, and JBoss Cache the Platform takes advantage of innovations in the open source community. As well, JBoss Enterprise Application Platform is fully tested and supported by Red Hat, and is certified to work on many leading enterprise hardware and software products.

2. NEW FEATURES IN JBOSS ENTERPRISE APPLICATION PLATFORM 5.0

JBoss Enterprise Application Platform 5.0.0 is a major release with many updated and additional components.

2.1. JBoss AS

The JBoss AS 5.1.x family represents the state of the art, second generation Microcontainer based enterprise Java run-time and the latest release of the world's most popular application server. In addition to supporting the latest Java EE specification (Java EE 5), it integrates many of the best enterprise class services for advanced messaging, persistence, transactions, caching and high-availability.

2.2. JBoss Microcontainer

The JBoss Microcontainer is a refactoring of the modular JBoss JMX Microkernel. It is a lightweight kernel that manages the loading, lifecycle and dependencies between POJOs. Used to integrate enterprise services, the JBoss Microcontainer together with Servlet/JSP container, EJB container, deployers and management utilities, provides a standard Java EE 5 profile.

2.3. JBoss Cache

JBoss Cache is used to replicate EJB and HTTP session state and supports distributed entity caching for JPA, while continuing to push the performance and scalability envelope with a new and more efficient locking scheme (MVCC – MultiVersion Concurrency Control).

2.4. JBoss Web Services

JBoss Web Services is a framework that supports the latest JAX-WS specification and a pluggble architecture to provide choice of Web Services Stack.

2.5. The Native Package

The Native package is an optional component for the JBoss Enterprise Application Platform and incorporates JBoss Native and the mod_jk technologies. These technologies are described as follows:

- JBoss Native consists of the Apache Portable Runtime (APR), OpenSSL and Tomcat Native (TC-native).

  - Apache Portable Runtime (APR): provides superior scalability, performance, and improved integration with native server technologies. The Apache Portable Runtime is a highly portable library that is at the heart of Apache HTTP Server 2.x and contains many uses, including access to advanced IO functionality (for example: sendfile, epoll and OpenSSL), Operating System level functionality (for example: random number generation and system status), and native process handling (shared memory, NT pipes and Unix sockets).

  - OpenSSL: implements the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols and includes a basic crytographic library.

  - Tomcat Native (TC-Native): is a Java Native Interface (JNI) that provides much of Tomcats core functionality in native code rather than Java. This allows for an overall increase in the speed of a server.

- mod_jk: is a connector that is used to connect the Tomcat JSP container and different web services such as Apache.

NOTE

The Red Hat Enterprise Linux 5 distribution of the Native package does not contain OpenSSL or the Apache Portable Runtime since these technologies are already included in the Operating System.

3. COMPONENT VERSIONS

This section details the versions of the components which create the Enterprise Application Platform 5.0.0.

Table 1. Component Version Differences

Component	EAP 5.0.0	EAP 4.3.0 CP06
JBoss Application Server	5.1.0.GA	4.2.0
JBoss Microcontainer	2.0.9.GA	1.0.2
JBoss Native	2.0.6.GA	N/A
Hibernate Core	3.3.2.GA	3.2.4.SP1.CP08
Hibernate Entity Manager	3.4.0.GA	3.3.2.GA
Hibernate Annotations	3.4.0.GA	3.3.1.GA_CP01

Component	EAP 5.0.0	EAP 4.3.0 CP06
Hibernate Search	3.1.1.GA	3.0.0.GA
Hibernate Validator	3.1.0.GA	3.0.0.GA
Clustered Remote EJB3 Proxies	1.1.18.GA	N/A
JBoss Cache	3.2.1.GA	1.4.1_SP13
JBoss HA Server API	1.1.1.GA	N/A
JBoss JAXR	2.0.1	1.2.0.SP2
RESTEasy	1.1.CP01	N/A
JGroups	2.6.13.GA	2.4.6.GA
JBoss EJB3	1.1.18	4.2-Snapshot (Technology Preview)
JBoss JTA	4.6.1.GA_CP03	N/A
JBoss JTS	4.6.1.GA_CP03	N/A
JBoss Negotiation	2.0.3.SP1	N/A
JBoss Managed	2.1.1.GA	N/A
JBoss Messaging	1.4.6.GA	1.4.0.SP3-CP08.patch01
JBoss Metadata	1.0.1.SP1	N/A
JBoss Web	2.1.3.GA	2.0.0.CP10
JBoss Web Services - Native	3.1.2.SP3	N/A
JBoss AOP	2.1.6.GA	1.5.5.GA_CP03
JBoss Remoting	2.5.2	2.2.3
JBoss Serialization	1.0.3.GA	1.0.3.GA
JBoss XB	2.0.1.GA	1.0.0.SP3
JavaServer Faces	1.2_13	1.2_10

Component	EAP 5.0.0	EAP 4.3.0 CP06
JacORB	2.3.1.jboss.patch01	2.3.0.jboss.patch6
JPA	1.0.0	N/A
JBoss Security	2.0.4.SP1	N/A
JBoss Profiler-jvmti	1.0.0.CR5	1.0.0.CR5
Seam	2.2.0.EAP5	1.2.1.AP
RichFaces (in Seam)	3.3.1.GA	N/A
JON JASA Console	1.3.2.GA	N/A
mod_jk	1.2.28	N/A
mod_cluster	1.0.2.GA	N/A

NOTE

JBoss Web Services CXF 3.1.2.SP2 is also provided as a Technology Preview.

4. INSTALLATION NOTES

4.1. Installing JBoss Enterprise Application Platform

You must have adequate disk space to install JDK and JBoss Enterprise Application Platform while also allowing enough space for your applications. You must have a working installation of JDK 1.6. For the latest information on supported Operating System / JVM combinations, supported Database platforms and current information on the revision level of included components, refer to the installation guide available online from http://www.redhat.com/docs/en-US/JBoss_Enterprise_Application_Platform/. The installation guide also details instructions to install and verify the installation of the JBoss Enterprise Application Platform.

4.2. Default Startup Profile

The default startup profile is `default` which is a base Java EE 5 server profile containing a default set of services. It contains the most frequently used services required to deploy a Java EE 5 application. It does not include the JAXR service, the IIOP service, or any of the clustering services.

The `default` profile is not intended for production use or for the running of load, stress, availability or performance tests.

4.3. Source Files

Source ZIP

● ftp://ftp.redhat.com/pub/redhat/jbeap/5.0.0/en/source/jboss-eap-src-5.0.0.GA.zip

5. PRODUCT SUPPORT

Bugs, potential bugs, and development issues and/or questions should be filed as JBoss support cases via the JBoss Customer Support Portal.

6. DOCUMENTATION

Refer to the index.html file in the documentation directory for a list of included documentation. In this release all API documents, code examples and a link to the online release notes are included with the distribution; all other guides and documentation are avaliable online at http://www.redhat.com/docs/en-US/JBoss_Enterprise_Application_Platform/.

In the ZIP, the included distribution for the Platform and its individual components can be found in a separate ZIP file, `jboss-eap-docs-<version>.zip`.

Amongst the online documentation you will find the following important guides:

● **Installation Guide** explains how to install and verify the installation of JBoss Enterprise Application Platform using different installation modes.

● **Getting Started** details the directory structure of the platform and a quick tour of the Application Server and different configuration sets and services.

● **Administration and Configuration Guide** explains all administrative and server configuration functions in detail.

The online documentation will be updated as necessary so be sure to check back, especially when a new version of the JBoss Enterprise Application Platform is released.

7. ISSUES FIXED IN THIS RELEASE

Following is a list of issues fixed in this release:

JBoss Application Server

● JBPAPP-2866: The JGroups protocol stack included an incorrect diagnostic address, `224.0.0.75`. The address has been corrected to `224.0.75.75`.

● JBPAPP-2306: The HP-UX init script in the First Customer Ship, `jboss_init_hpux.sh`, did not let users define a bind address with the `-b` option. Support for this option has since been added to the script.

JBoss Messaging

● JBPAPP-2924: Between the First Customer Ship and General Availability releases of JBoss Enterprise Application Platform 5.0, the included version of JBoss Messaging was upgraded from 1.4.3 to 1.4.6 to provide better support for JBoss Enterprise Application Platform 5.0. For further information about the differences between JBoss Messaging 1.4.3 and JBoss Messaging 1.4.6, check the JBoss Messaging Release Notes.

JBoss Seam

- **JBPAPP-2154**: Some unsupported components such as IceFaces, Seam-jul and Wicket integration were included in the First Customer Ship. These unsupported components and the examples related to them have been removed.

- **JBPAPP-2155**: The Seam documentation included in the First Customer Ship contained unverified and unsupported information about non-JBoss containers. Unsupported information has been removed from the documentation, and the examples to do with unsupported containers have been removed.

JBoss Hibernate

- **JBPAPP-2900**: MySQL uses the **TEMPORARY** keyword to bypass implicit transaction commits. Previously, Hibernate used **<CREATE TEMPORARY TABLE>** with **<DROP TABLE>**. Omitting the **TEMPORARY** keyword caused an implicit commit, and immediate failure within an XA Transaction. **<DROP TEMPORARY TABLE>** is now supported and this issue no longer presents.

Documentation

- **JBPAPP-2948**: The **deploy/jmx-remoting.sar** service instantiates a JSR-160 adapter for standardized remote access to the JBoss MBeanServer. This service is used with tools such as the JConsole. At present, this service does not support secure access. In production environments where the server binds to a specific address other than **localhost** this presents a potential security risk, so the adapter has been moved from the **deploy** directory into **docs/examples/jmx**. We do not recommend enabling it for production usage. If during development you wish to re-enable the adapter, copy it back to the **deploy** directory.

  The adapter has been moved to **/docs/examples**. If you wish to re-enable it, move it back to the **deploy** directory.

8. KNOWN ISSUES WITH THIS RELEASE

Following is a list of known issues at the time of release.

General Known Issues

- **JBPAPP-3036**: The **jboss_init_hpux** script does not pick up environment variables when executed in the GNU bash shell. This is related to JBPAPP-2036: https://jira.jboss.org/jira/browse/JBPAPP-2306.

- **JBPAPP-3035**: The **-e** and **-H** arguments of the **shutdown.sh** script cannot be used to directly terminate the JVM.

- **JBPAPP-3029**: The **jboss_init_redhat.sh** script is used to start and stop a server instance under a given user name. When using a non-loopback bind address, calling **jboss_init_redhat.sh stop** results in a **CommunicationException** because of a missing hostname parameter for the remote server the script attempts to contact.

- **JBPAPP-2998**: When the server is started via a desktop icon, the machine's default Java set is used. This can cause exceptions when a Java version other than JDK 1.6 is used.

- **JBPAPP-2571**: Running the Microsoft SQL server with Microsoft JDBC drivers 2.0 causes unstable builds in JBoss Messaging. This issue is created by the **Adaptive Buffering** being the default behavior for the driver as this only allows for large values ot be read once by using

the `get<Type>Stream` method. The current workaround for this issue is to change the *responseBuffering* parameter of the JDBC connection URL from being `adaptive`, to being `full` (as it was for version 1.2 of the JDBC driver).

```
<url>jdbc:sqlserver://[host];database=
[database];responseBuffering=full;</url>
```

- JBAS-7049: The server manager does not function correctly when Open JDK 6 is used because a `NullPointerException` check is missing in Open JDK 6. The workaround is to comment out the java.security.debug statement in the `imports/server-config.xml` file.

- JBPAPP-2598: Once the workaround for the JBAS-7049 issue is applied a new issue presents itself. A server running the surity manager using Open JDK 6 still fails to start, now with an access denied error. There is no currently known workaround for this issue.

- JBPAPP-2590: An issue exists in the `policy.provider` defined in `${JAVA_HOME}/jre/lib/security/java.security` when IBM JDK 6 is used. By default `org.apache.harmony.security.fortress.DefaultPolicy` is used and this should be `policy.provider=sun.security.provider.PolicyFile`. The workaround for this issue is to manually make this adjustment.

- JBPAPP-2576: The MySQL JDBC driver does not currently implement XA Recovery correctly.

- JBPAPP-2871: High CPU utilization and reduced performance and transaction throughput has been observed when MySQL 5.0.41 when optimized settings are in use, as described in JBQA-2610. We recommend upgrading to MySQL 5.0.86 and applying optimized settings as described in JBQA-2610 to reduce CPU utilization and increase performance.

- JBPAPP-2162: Sun JAXB silently accepts messages with non-fatal errors where they should be rejected. The fix for JBPAPP-2114 corrects this, so that badly-formed messages are rejected.

- JBPAPP-2765: `LoadMgr3` logs a failure to load a class as an error even when the failure to load is expected and intentional. (Seam, for example, catches exceptions in order to disable unnecessary components if a particular class is not found.)

- JBPAPP-2894: Setting the `hibernate.bytecode.provider` system property in `jpa-deployers-jboss-beans.xml` is unreliable. The workaround for this issue is to add `-Dhibernate.bytecode.provider=cglib` to $JAVA_OPTS in `jboss-as/bin/run.conf`.

- JBPAPP-2818: The current `main/src/bin/run.sh` does not allow users to override `$JBOSS_HOME/bin/run.conf` with a profile-specific `$JBOSS_HOME/server/$PROFILE/run.conf`.

- JBPAPP-2713: `org.jboss.test.xml.DDValidatorUnitTestCase` constantly fails and crashes the Java Virtual Machine. The workaround for this issue is to disable the JIT compiler by setting `JAVA_COMPILER=NONE` or using the command line switch `-Djava.compiler=NONE`.

- JBAS-6966: The IBM distribution of JDK 6 does not support the `SSLv2Hello` protocol and generates a ERROR [AbstractKernelController] when used. It is currently recommended not to use this protocol.

Hibernate Known Issues

- **JBPAPP-3034**: When batch insert statements are ordered, embedded classes are not taken into account. There are two possible workarounds for this issue. The first is to set **ORDER_INSERTS** to **FALSE** when embedded classes are used. The second option is to explicitly call **session.save()** on child objects to enforce their SQL insertion orders.

- **JBPAPP-3032**: MySQL does not currently support millisecond and microsecond measurements when returning database values such as **TIME** and **TIMESTAMP**.

- **JBPAPP-3031**: On Sybase, the **current_timestamp** text is not being recognized by the translator as a *method mode*. There is currently no way to work around this issue except to avoid relying upon function replacement for **current_timestamp**.

- **JBPAPP-3030**: On Sybase, **SchemaExport** cannot be used to create stored procedures while in while in chained transaction mode. The workaround for this case is to add the following code immediately after the definition of the new stored prodecure:

```
<database-object>
<create>
sp_procxmode paramHandling, 'chained'
</create>
<drop/>
</database-object>
```

- **JBPAPP-2791**: Applications that map Hibernate to use **cglib** as a byte provider fail to deploy because of a **java.lang.SecurityException**. An error message similar to the following is displayed:

```
Deployment "persistence.unit:unitName=lobtest.ear/lobtest-ejb-1.0-
SNAPSHOT.jar#lobtest-jpa-jndi" is in error due to the following
reason(s): java.lang.SecurityException: class
"com.redhat.gss.lobtest.jpa.Item$$EnhancerByCGLIB$$defd1a7f"'s
signer information does not match signer information of other
classes in the same package
```

This occurs because the **cglib.jar** in JBoss Enterprise Application Platform 5.0 is signed, and the cglib-instrumented proxy uses the **cglib.jar** signer information instead of the signer information of the application target class.

The patch for this issue has been released alongside JBoss Enterprise Application Platform 5.0 and can be downloaded from Red Hat Support.

- **JBPAPP-2957**: The **evictAll()** method in **EntityRegionAccessStrategy** and **CollectionRegionAccessStrategy** attempts to remove objects from the cache without regard for transaction isolation. Disregarding transaction isolation is currently unsupported because JBoss Cache is allowed visibility to any active transaction and will therefore use the transaction to control access to data in the cache.

- **JBPAPP-2945**: Setting the query timeout for a PreparedStatement is not supported by PostgreSQL 8.3.7. This limitation means that queries will fail if they use an annotation like the following:

```
@QueryHint(name = "org.hibernate.timeout",
value = "100")
```

- JBPAPP-2892: When Enterprise JavaBean 3.0 entities are used with optimistic caching, `org.jboss.ejb3.entity.OptimisticJBCCache.DataVersionAdapter.newerThan` incorrectly returns `true` for `A.newerThan ( A )`. This causes a `DataVersioningException` when JBoss Cache attempts to remove the entry. The workaround for this issue is to use Multiversion Concurrency Control (`mvcc-entity`) instead of optimistic caching. This is recommended even if the exception has not been observed.

- JBPAPP-2867: Sybase does not currently support Hibernate `Blob`s or `Clob`s, and Hibernate does not support Sybase `text` or `image` data types. The workaround for this issue is to create user-defined types that map to the Sybase `text` and `image` types.

- JBPAPP-2789: **ShemaExport** fails on Oracle and Sybase when a redundant `@Column(unique=true)` or `UniqueContraint(columnnames={...})` annotation is used on a column that is implicitly defined as unique by the declared model. The workaround is to remove the redundant `@Column(unique=true)` or `UniqueContraint(columnnames={...})` annotation.

- JBPAPP-2613: When the DB2 v9.7 driver is used with progressive streaming (the default), operations on `Blob` and `Clob` locators fail. The workaround is to disable progressive streaming using connection properties, as recommended in the linked JIRA issue.

- JBPAPP-2408: An issue exists with the DB2 v9.7 driver when using an identity or native ID generator with Hibernate. The `Statement.getGeneratedKeys()` driver method in DB2 returns an empty *resultset* instead of the generated keys, which causes Hibernate to throw an exception that states that the database returned no natively generated identity value. This issue has been fixed in the version of DB2 9.7 JDBC driver released with Data Studio 2.2 and is available for download from the DB2 website. This is the recommended version for use with Hibernate.

- JBPAPP-2278: The **Save** operation may fail when a transient entity is reachable by multiple paths and at least one of those paths does not cascade for the **Save** opperation. The current workaround for this is to save the transient entity before executing the save that had previously failed. If this is not possible, another workaround is to modify either or both cascade and entity mappings to change the order of the cascade paths so that the transient entity is saved before it cascades to the entity that requires it to be non-transient.

- JBPAPP-2277: Hibernate uses `ClassLoader.loadClass()` for arrays, which is no longer supported by default as of Java 6. The current workaround for this issue is to use the -Dsun.lang.ClassLoader.allowArraySyntax=true property. This workaround is implemented by default as the default in `jboss-eap-5.0/jboss-as/run.conf`.

- JBPAPP-2276: The iteration order of `HashMaps` and `HashSets` for JDK 6 causes the order of columns in union clauses or union subclasses to differ depending on whether JDK 5 or 6 is used. Since the change in column order is consistent across union clauses, the resulting queries are valid; however, this change can potentially affect performance.

- JBPAPP-2275: Hibernate cannot be compiled under JDK 6. This occurs because the following classes require methods to be added in order to fully implement JDK 6 interfaces:

  - `org.hibernate.jdbc.ResultSetWrapper`

  - `org.hibernate.lob.BlobImpl` to implement `java.sql.Blob`

  - `org.hibernate.lob.ClobImpl`

  - `org.hibernate.lob.SerializableBlob`

- `org.hibernate.lob.SerializableClob`

If the application you are running requires a method missing from the above classes, a NoSuchMethodError will be generated.

- JBPAPP-2922: Hibernate warns that **The CGLIB BytecodeProvider impl is considered deprecated and not recommended for use.** `cglib` is not deprecated, so this warning can be safely ignored.

- JBPAPP-2792: Sybase fails to insert a new entity if it overflows its column. However, it does not throw an exception, so Hibernate cannot tell that the insert failed. To work around this issue, the application must validate entity properties to ensure that they do not overflow the underlying column.

- JBPAPP-2791: **SchemaUpdate**s fail in Sybase ASE 15 tables when a new column is added without a default value. To work around this issue, ensure that a default value is included when adding new columns with **SchemaUpdate**.

- JBPAPP-1613: Null values for columns mapped as **Boolean** in Sybase are persisted as **0** instead of **null**. The workaround for this issue is to map **type="org.hibernate.test.where.NumericTrueFalseType"** instead of **type="boolean"**.

- JBPAPP-1554: Sybase only allows only one entry (for example, column name or '*') in a subquery select list. The HQL function, **elements()**, fails when the collection elements have a composite ID, because the generated SQL contains a subquery select list with multiple entries. The workaround is to avoid using HQL **elements()** if the elements have a composite key. Instead, reformulate the HQL so that no subquery has multiple entries in its select list.

- JBPAPP-1547: **org.hibernate.dialect.SybaseASE15Dialect.areStringComparisonsCaseInsensitive()** returns **false** and is not overridden by **SybaseDialect**.

- JBPAPP-1545: On Sybase, when a query has an ANSI join with three or more joins, and one join involves a union subclass, the query may fail with **SybSQLException** because a column is not within the scope of the joined table expression. The current recommendation is to avoid using join fetches that involve union subclasses.

- JBPAPP-2858: **getSingleResult()** fails for some databases and queries when automatic pagination is attempted. The workaround for this issue is to use **getResultList()** instead of **getSingleResult()**.

- JBPAPP-1998: **EntityNotFoundException** is incorrectly thrown upon an optimistic locking failure when one **EntityManager** tries to delete an entity that has been updated by a different **EntityManager** and **hibernate.jdbc.batch_versioned_data** is set to **false** (the default value). **OptimisticLockException** should be thrown instead.

- JBPAPP-1230: When a **DetachedCriteria** is used as a subquery, the generated SQL contains a column alias in the subquery. On Sybase, a **SybSQLException** is thrown because Sybase does not allow column aliases in subqueries. The workaround for this issue is to use an HQL query instead of a **DetachedCriteria** in a subquery.

- JBPAPP-1123: When @**OrderBy** is used on joined classes (using a join table), the generated SQL is invalid on MySQL, PostgreSQL, Oracle, and MSSQL because the "order by" clause qualifies the columns using the actual table name. The "order by" clause should use the table

alias instead.

- JBPAPP-1082: When a **char** property is used and it is not initialized, then Hibernate initializes it to **0** and persists a string containing the character **\u0000**. PostgreSQL throws an exception because it does not allow character **\u0000** embedded in a string. An email thread describing this problem is available from:

  - http://markmail.org/message/jvzlzxw6ui2rcsmv#query:\u0000%20postgresql+page:1+mid

  There is currently no workaround for persisting **\u0000** in a **char** column using PostgreSQL.

  If the user intends to persist a NULL for an uninitialized **char** property instead of **\u0000**, then **java.lang.Character** should be used instead of the primitive **char** type. This avoids the exception when the property is initialized. Attempting to persist a **java.lang.Character** property that is set to **\u0000** still results in an exception.

- JBPAPP-1071: In some cases, when there are foreign key constraints defined on columns in a primary key, **SchemaExport** incorrectly declares them as nullable when it generates **CREATE TABLE** statements. This fails on MSSQL, DB2, and Sybase because those databases require primary key columns to be non-nullable.

  The workaround for this issue is to explicitly indicate which columns should be non-nullable, such as:

  - Add **nullable=false** to **@JoinColumn**

  - Add **optional=false** to **@ManyToOne**

  - Add an **@AttributeOverride** with **@Column(name="mapkey", nullable=false)** in case of a **@CollectionOfElements** using a Map

  - Add **nullable=false** in **@Column** when inside a **@CollectionId** or inside **@MapKey**

- JBPAPP-3010: The **evict(Object)** method in **EntityRegionAccessStrategy** and **CollectionRegionAccessStrategy** attempts to remove objects from the cache without regard for transaction isolation. This is currently unsupported because JBoss Cache's **removeNode** method does not deal with transactions.

- JBPAPP-3019: The **doc/examples/jboss-web-services-examples** context causes a number of exceptions to occur. This context error means that the JBoss Web Services examples will not work correctly.

JBoss Messaging Known Issues

- JBPAPP-3023: The version of JBoss Messaging included in JBoss Enterprise Application Platform 4.3 allowed users to declare message queues with heirarchical names such as **app1/emails**, where the forward slash would create a nested JNDI context of **queueu/app1/emails**. This behavior is not currently supported in JBoss Enterprise Application Platform 5.0. There are two possible workarounds for this issue:

  1. Users can create the queue or topic in the configuration files instead of relying on auto-creation. (This is recommended.)

  2. Users can change the **destination** attribute (which represents the JNDI name of the destination) so that it is non-heirarchical, for example, from **queue/somequeue** to **somequeue**.

- **JBPAPP-1745**: When JBoss Transactions is upgraded from version 4.4.0 to version 4.6.1, the JBoss Messaging **XAResourceRecoveryTest** fails. The test is meant to simulate a failure during the commit process to prompt the Recovery Manager to recover the transaction, but with JBoss Transactions 4.6.1, the transaction is not recovered. The workaround for this failure is to manually start the Recovery Manager with the following code:

  ```
              recoveryManager =
  RecoveryManager.manager(RecoveryManager.INDIRECT_MANAGEMENT);
  ```

- **JBPAPP-1746**: When JGroups is upgraded to version 2.6.13, the JBoss Messaging **LargeClusterTest** fails. The test attempts to start seven nodes on one machine before testing, but some nodes fail to start. The test passes if the number of nodes is reduced.

- **JBPAPP-2033**: **EnableMessageCounters** on the **jboss.messaging.ServerPeer** JMX interface, cannot be set to **true**. A workaround to enable message counters is to invoke the **enableMessageCounters()** operation on the same JMX interface.

JBoss Web Services Known Issues

- **JBPAPP-3028**: The upgrade from JBoss Web Services 2.0.1.SP to JBoss Web Services Native 3.1.2.SP incorporated many changes and new features. The additional processing time required for several new features (resource injection, support for **@PostConstruct** and **@PreDestroy**, etc.) has resulted in a slight degradation in overall performance.

mod_cluster Known Issues

- **MODCLUSTER-113**: **org.jboss.modcluster.demo.servlet.ThreadCountLoadServlet** has been removed from mod_cluster but is still specified in the **web.xml** file belonging to **load-demo.war**. This results in deployment errors. The workaround for this issue is to remove the **<servlet>** and **<servlet-mapping>** sections for the **threads** servlet.

-

RESTEasy Known Issues

- **JBPAPP-2991**: The readme file for the API Clients example located in **doc/examples/resteasy-examples/api-clients** contains an invalid command for the Twitter example startup:

  ```
  Twitter Client:
  - Run:
      mvn exec:java -
  Dexec.mainClass="org.jboss.resteasy.examples.twitter.Twitter" -
  Dexec.args="<userId> <password>"
  ```

  Replace the previous code with the following:

  ```
  Twitter Client:
  - Run:
      mvn exec:java -
  Dexec.mainClass="org.jboss.resteasy.examples.twitter.TwitterClient"
   -Dexec.args="<userId> <password>"
  ```

`doc/examples/resteasy-examples/api-clients/` also contains two superfluous Eclipse project files, `.classpath` and `.project`. We recommend deleting these files.

- JBPAPP-2992: the readme file for the Spring MVC example located at `doc/examples/resteasy-examples/resteasy-springMVC/README.txt` contains an invalid URL. The current text:

      List all available contacts:
      http://localhost:9095/contacts

  Should be replaced with the following:

      List all available contacts:
      http://localhost:8080/contacts

Seam Known Issues

- JBPAPP-2385: Seam's Spring example fails with an `IllegalStateException` when the login form is submitted on the IBM virtual machine. This is due to a defect in the IBM virtual machine. The fix for this issue has been deferred until the IBM virtual machine has been fixed.

- JBPAPP-2377: The Seamspace example fails with a `NullPointerException` when submitting a new blog entry on the IBM virtual machine. This is due to a defect in the IBM virtual machine. The fix for this issue has been deferred until the IBM virtual machine has been fixed.

- JBPAPP-3001: The seam script included in this distribution does not have an executable permission set. An executable permission set is planned for inclusion in the next release of JBoss Enterprise Application Platform 5.

9. TECHNOLOGY PREVIEWS

This section describes the Technology Preview features released alongside JBoss Enterprise Application Platform, their installation, and any known issues.

> **WARNING**
>
> Technology Preview features are not fully supported under Red Hat subscription level agreements (SLAs), may not be functionally complete, and are not intended for production use. However, these features provide early access to upcoming product innovations, enabling customers to test functionality and provide feedback during the development process. As Red Hat considers making future iterations of Technology Preview features generally available, we will provide commercially reasonable efforts to resolve any reported issues that customers experience when using these features.

9.1. JBoss Web Services CXF

> **WARNING**
>
> You should make a complete backup of your JBoss Enterprise Application Platform installation before installing JBoss Web Services CXF.

Follow these steps to install JBoss Web Services CXF:

1. Download and unzip the `jbossws-cxf-3.1.2.SP3-3.SP3.6.ep5.el4-tp-installer.zip` in the home directory (your `$JBOSS_HOME`) of a pristine Enterprise Application Platform installation.

2. Run `ant` in the created directory, `jbossws-cxf-tp-installer`.

   > **NOTE**
   >
   > This step will replace JBoss Web Services Native with JBoss Web Services CXF in every configuration that contains JBoss Web Services Native.

Documentation is available from the JBoss.org Wiki: http://community.jboss.org/wiki/JBossWS-StackCXFUserGuide.

9.1.1. Known Issues with JBoss Web Services CXF

- JBPAPP-3267: Running `ant deploy-jboss510` from the stack checkout directory or a generated binary distribution does not result in a working application server configuration. This occurs primarily as a result of changes to `src/main/scripts/assembly-deploy-artifacts.xml`, which were made to produce the `tp-installer`.

9.2. JBoss mod_cluster

`mod_cluster` is an HTTPD-based load balancer. Like `mod_jk`, it uses a communication channel to forward requests from HTTPD to an application server node.

See the mod_cluster documentation available from http://www.redhat.com/docs/en-US/JBoss_Enterprise_Application_Platform/ for installation instructions.

9.2.1. Known Issues with JBoss mod_cluster

- JBPAPP-3463: When an application is undeployed, a race condition occurs between session invalidation and application stop. The session becomes invalid before application stop occurs, leading to a 503 - This application is not currently available .

A. REVISION HISTORY

Revision 5.0.0-12.400   2013-10-31   **Rüdiger Landmann**
   Rebuild with publican 4.0.0

Revision 5.0.0-12   2012-07-18   **Anthony Towns**
   Rebuild for Publican 3.0

Revision 1.9-0   Mon Jan 25 2010   **Laura Bailey**
   JBoss mod_cluster added.