



# **JBoss Enterprise Application Platform 5**

## **Release Notes 5.2.0**

for use with JBoss Enterprise Application Platform 5.2.0

Edition 5.2.0



# JBoss Enterprise Application Platform 5 Release Notes 5.2.0

---

for use with JBoss Enterprise Application Platform 5.2.0

Edition 5.2.0

Eva Kopalova

Petr Penicka

Russell Dickenson

Scott Mumford

## Legal Notice

Copyright © 2012 Red Hat, Inc.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

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, OpenShift, 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.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

These release notes contain important information related to JBoss Enterprise Application Platform 5.2.0 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 5.2.0.

---

## Table of Contents

<b>CHAPTER 1. INTRODUCTION</b> .....	<b>3</b>
1.1. ABOUT JBOSS ENTERPRISE APPLICATION PLATFORM	3
1.2. ABOUT JBOSS ENTERPRISE WEB PLATFORM	3
1.3. JBOSS ENTERPRISE APPLICATION PLATFORM AND JBOSS ENTERPRISE WEB PLATFORM COMPONENTS	3
1.4. ABOUT THIS RELEASE	3
1.5. NEW FEATURES	4
1.6. EXCLUDED, REMOVED, OR DEPRECATED ITEMS	4
1.6.1. Deprecated	4
<b>CHAPTER 2. FEATURE REQUESTS</b> .....	<b>6</b>
<b>CHAPTER 3. KNOWN ISSUES</b> .....	<b>9</b>
<b>CHAPTER 4. RESOLVED ISSUES</b> .....	<b>13</b>
<b>APPENDIX A. REVISION HISTORY</b> .....	<b>38</b>
<b>INDEX</b> .....	<b>39</b>



# CHAPTER 1. INTRODUCTION

These release notes contain important information related to JBoss Enterprise Application Platform 5.2.0 and JBoss Enterprise Web Platform 5.2.0. New features, issues fixed in this release, and other known issues are addressed here.

## 1.1. ABOUT JBOSS ENTERPRISE APPLICATION PLATFORM

JBoss Enterprise Application Platform is an open source platform for development, deployment, and management of Web 2.0 applications.

The product integrates open source frameworks such as JBoss Seam, Hibernate, CXF Web Services, JBoss Cache, and JBoss Messaging. JBoss Enterprise Application Platform is fully tested and supported by Red Hat, and is certified to work on numerous leading enterprise hardware and software products.

## 1.2. ABOUT JBOSS ENTERPRISE WEB PLATFORM

JBoss Enterprise Web Platform is a slimmed-down version of JBoss Enterprise Application Platform compliant with the Java EE Web Profile specification. In addition to the specification, it provides enterprise features for clustering, caching, and simple web services. This product is intended for light and rich Java applications and to handle mid-size workloads.

## 1.3. JBOSS ENTERPRISE APPLICATION PLATFORM AND JBOSS ENTERPRISE WEB PLATFORM COMPONENTS

Since JBoss Enterprise Web Platform is a slimmed-down version of JBoss Enterprise Application Platform, only some of the release notes are relevant for both products. The release notes of the following components are specific to JBoss Enterprise Application Platform:

- Hibernate Annotations
- Hibernate Search
- JBoss WS-CXF
- JacORB
- Xerces
- JAXB
- JBoss Messaging
- HornetQ
- mod\_cluster
- PicketLink Federation
- Netty

## 1.4. ABOUT THIS RELEASE

JBoss Enterprise Application Platform 5.2.0, while not a major release, is nonetheless a non-trivial iteration of the product.

This release aggregates the contents of prior patches and Cumulative Patches (CPs), and adds functionality and updated components. Subsequent patches and Cumulative Patches will assume the installation of this update.

During the life cycle of a product's major version, Red Hat makes commercially reasonable efforts to maintain API-level compatibility across all minor releases and asynchronous patches, ensuring that, for example, JBoss Enterprise Application Platform 5.2.0 maintains API-level compatibility with JBoss Enterprise Application Platform 5.0.0, the initial release of JBoss Enterprise Application Platform 5. Possible exceptions to this rule include fixes addressing critical security issues.

JBoss Enterprise Application Platform 5 customers are advised to update to JBoss Enterprise Application Platform 5.2.0.

However, since the release of Enterprise Application Platform version 5.1.1 it is not possible to perform an upgrade from version 4.3. Specifically, updating to 5.2.0 from 4.3 using RPMs is not supported, and will result in platform failure.

Refer to [http://www.redhat.com/security/updates/jboss\\_notes/](http://www.redhat.com/security/updates/jboss_notes/) for more information about the *JBoss Enterprise Middleware Product Update and Support Policy*.

## 1.5. NEW FEATURES

This release adds certification for the following third-party components:

- IBM WebSphere MQ
- Java 7

## 1.6. EXCLUDED, REMOVED, OR DEPRECATED ITEMS

### Definitions

#### Excluded

An item that has never been featured in a product release but is otherwise part of one of the open source components of the product.

#### Deprecated

An item that will be removed from a future release, usually the next major version.

#### Removed

An item that was previously in a release of the product and is no longer included. Items will usually be deprecated before being removed.

### 1.6.1. Deprecated

#### JBAS-7837

JBoss Cache is deprecated as it is to be replaced by Infinispan in the next major release.

#### EJBTHREE-2173



The `hibernate-jmx` JAR is deprecated and will be removed from `jboss-ejb3-as6-depchain` in the next major release. The `hibernate-annotations` JAR will be reimplemented and delivered as part of the `hibernate-core` JAR.

#### **JBAS-7527**

The JDBC3 (JDK5) classes from the JDBC adapter are deprecated and will be removed in the next major release as a JDK6 (or later) environment, which contains these classes, will be required.

#### **JBAS-7563**

As `RMIAdaptor` is being replaced to meet JSR-160, the `JRPMInvoker` and `JRMPInvokerHA` are deprecated and the `UnifiedInvoker` will be the default legacy invoker in the next major release.

#### **JBPAPP-3961**

JBoss Web Console is deprecated in JBoss Enterprise Application Platform 5.2. The Admin Console will take over its function. Also note that JMX Console will be supported as legacy. Note: The Web Console is not included at all in JBoss Enterprise Application Platform 6.

#### **JBPAPP-4995**

The `TwitterClient` example is deprecated in `RESTEasy 1.2.x` due to Twitter deprecating the Basic Authentication method in August 2010. All applications must now use OAuth. `RESTEasy 2.x` contains a reworked `TwitterClient` example that includes OAuth. Download the example from the newest version of `RESTEasy` <http://www.jboss.org/resteasy> for testing purposes.

## CHAPTER 2. FEATURE REQUESTS

Feature requests in JBoss Enterprise Application Platform 5.2.0 are listed by component.

### Clustering

<https://issues.jboss.org/browse/JBPAPP-9167>

The TCPPING configuration in JGroups had to statically define the list of all possible cluster members. Consequently, the cluster had to be restarted if a new member was added. The `max_dynamic_hosts` property has been added to TCPPING which defines the number of extra hosts to be added dynamically to the cluster without the need to restart the cluster. Therefore, it is now possible to add new members to the cluster when using TCPPING without requiring the cluster restart. The property default value is 0. Note that the new nodes need to be added statically to each node's configuration at the next restart of that node.

<https://issues.jboss.org/browse/JBPAPP-9370>

When JGroups caught an `IOException` while creating server sockets, the root exception was hidden and not included in the thrown exception. This way, users could not identify the cause of the error that occurred. The root exception is now included in the thrown exception so that users can identify the root cause of the error.

<https://issues.jboss.org/browse/JBPAPP-9562>

When a cluster member does not respond before timeout, other members can fail to join the cluster. Previously, in this case, the logs contained only an entry about the timeout event that occurred while waiting for responses from the cluster ("GMS flush by coordinator at <ip\_address>: <port> failed ") and did not report which members were not present in the cluster. Now, the log level of this event was changed from DEBUG to WARN, and the log event contains a list of the missing cluster members to allow easier troubleshooting.

### Documentation

<https://issues.jboss.org/browse/JBPAPP-8223>

The *Installation Guide* did not explicitly state that the latest security patches should be applied immediately after installation. This has now been added to the guide immediately following the installation instructions.

<https://issues.jboss.org/browse/JBPAPP-9496>

The *Getting Started Guide* has been restructured and simplified.

<https://issues.jboss.org/browse/JBPAPP-5299>

Information about how to enable LDAP integration with JBossWS has been added to the *WS CXF User Guide* book.

<https://issues.jboss.org/browse/JBPAPP-5417>

A section describing `RoleMappingLoginModule` and its configuration options has been added to the *Security Guide*.

<https://issues.jboss.org/browse/JBPAPP-9299>

Information on how to remove `jsessionid` numbers from an application's URL has been added to the *Administration and Configuration Guide*

## HornetQ

<https://issues.jboss.org/browse/JBPAPP-8436>

The messages export tool has been rewritten to support large messages in HornetQ. It can now export and import both the journal and large messages' content.

## JCA

<https://issues.jboss.org/browse/JBPAPP-8119>

The JMS Java Connection Architecture's Resource Adapter has been enhanced to support `javax.jms.ConnectionFactory` and `javax.jms.XAConnectionFactory`.

## Scripts and Commands

<https://issues.jboss.org/browse/JBPAPP-8503>

The `jarcheck.sh`, `jarcheck.bat` and `jarcheck.jar` scripts have been added to the `JBOSS_HOME/bin` directory. These scripts collect MD5 checksums of system JAR files to identify which of them have been changed after installation.

<https://issues.jboss.org/browse/JBPAPP-7454>

JBoss Enterprise Application Platform attempted to load a per-profile `run.conf` file from the default location in `$JBOSS_HOME/server/$PROFILE`. Consequently, per-profile `run.conf` files stored in a different location were not loaded. The `--run-conf=...` option (on Windows: `-r`) has been added to the `run.sh` script, allowing to specify the location of the `run.conf` file that could be used. Using this option, per-profile `run.conf` files stored in another than the default location can now be loaded correctly.

## Security

<https://issues.jboss.org/browse/JBPAPP-8140>

An enhancement has been made to JBoss Negotiation when configured as an authentication source. If a client authenticates to the server using a forwardable Kerberos ticket then the client's credentials can be accessed on the server using the JBoss Negotiation API and used for further calls (delegation using GSSAPI, WS-Kerberos, credential propagation for EJB calls, etc.).

<https://issues.jboss.org/browse/JBPAPP-5882>

Exposing the keystore or truststore password in `JaasSecurityDomain` is insecure. The `getKey` and `getCertificate` methods have been added to retrieve private keys and certificates directly from the JSD so that they can be used by external components. The `getKey` method requires you to provide a security token.

## Web Services

<https://issues.jboss.org/browse/JBPAPP-7977>

In certain scenarios, JBossWS sends a large number of small packets over the network instead of sending a smaller number of larger packets. The following system property has been introduced in this release:

`-Dorg.jboss.ws.domwriter.FlushOnlyOnce=true`

If the value of the property is set to `true`, the size of packets sent over the network is increased and their number reduced.

<https://issues.jboss.org/browse/JBPAPP-7452>

JBossWS-CXF is used as the native stack for rewriting the `soap:address` values in WSDL definitions, except when a `jbossws-cxf.xml` file is provided in the web service's deployment. This enhancement allows the use of JBossWS-CXF even with the `jbossws-cxf.xml` file, by providing CXF-specific configuration in the file.

<https://issues.jboss.org/browse/JBPAPP-7441>

Web Services have been enhanced to allow wrapped style code generation with the `xsd:extension` of a wrapper-compatible type.

## CHAPTER 3. KNOWN ISSUES

Known issues in JBoss Enterprise Application Platform 5.2.0 are listed by component.

### Hibernate

<https://issues.jboss.org/browse/JBPAPP-5965>

You are not allowed to initialize lazy collections or proxies during a flush. The workaround is for the listener to use a different session to initialize the collection. The following is an example of the workaround:

```
SessionImplementor si = (SessionImplementor)(event.getSession());
Session anotherSession =
si.getFactory().openSession(si.getJDBCContext().connection());
Object obj = anotherSession.get(
event.getEntity().getClass(), event.getId());
if(o instanceof Parent){
    Parent parent = (Parent)obj;
    Iterator it = parent.getChildren().iterator();
    while(it.hasNext()){
        Child child = (Child)it.next();
    }
}
anotherSession.close()
```

<https://issues.jboss.org/browse/JBPAPP-6579>

Hibernate can bind the SessionFactory to JNDI after startup if it can find the session factory name from the server configuration. This results in a warning-level message, which is misleading in its severity. This message is unavoidable as it is usually caused by an environment-specific issue. To avoid seeing this message, adjust the logging level to **debug**.

<https://issues.jboss.org/browse/JBPAPP-4731>

When an insert operation was performed using `jconn3.jar`, `null` bit values were converted to `0` and no exception was thrown. Sybase does not allow a `null` value for the `bit` data type, so `null` values should result in exceptions.

<https://issues.jboss.org/browse/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.

<https://issues.jboss.org/browse/JBPAPP-9059>

When using Hibernate with IBM DB2-97 a `SQLException` may occur when deploying `jbossesb.esb`. The error occurs because of a mismatch between the `Hibernate` data type and the `DB2` data type to which it is mapped. There are two possible workarounds available for this issue:

- Add

```
<connection-property name="deferPrepares">false</connection-
property>
```

-

to `/jboss-as/server/db2-97/deploy/jbossesb-registry.sar/juddi-ds.xml`.  
 Note that according to <http://publib.boulder.ibm.com/infocenter/db2luw/v8/index.jsp?topic=/com.ibm.db2.udb.doc/ad/c0010273.htm>, this workaround may affect performance.

- Use `org.hibernate.type.NumericBooleanType` type instead of `BooleanType`.

<https://issues.jboss.org/browse/JBPAPP-6571>

A `NullPointerException` is thrown when an entity with attributes mapped as `@Formula` is obtained using the `addEntity` method by a native query whose select clause elements do not have an alias:

```
@Entity
@Table(name = "ALL_TABLES")
public class AllTables {
    @Id
    @Column(name = "TABLE_NAME", nullable = false)
    private String tableName;
    @Formula(value = "(SYSDATE())")
    private String daysOld;
    //getters and setters
}
```

```
String sql = "select table_name , sysdate() from all_tables where
table_name = 'AUDIT_ACTIONS' ";
SQLQuery q = s.createSQLQuery( sql ).addEntity( "t",AllTables.class );
```

To resolve this issue would require major changes to Hibernate, so this issue will remain unresolved. Use one of the following workarounds to avoid this issue:

- Workaround option 1: Use `@SqlResultSetMapping`:

```
String sql = "select table_name as t_name, sysdate() as t_time
from all_tables where table_name = 'AUDIT_ACTIONS' ";
@SqlResultSetMapping(name = "all",
entities = @EntityResult(entityClass = AllTables.class,
fields = {
@FieldResult(name = "tableName", column = "t_name"),
@FieldResult(name = "daysOld", column = "t_time")
}))
```

- Workaround option 2: Provide an alias for each scalar property:

```
String sql = "select t.table_name as {t.tableName}, sysdate() as
{t.daysOld} from all_tables t where t.table_name = 'AUDIT_ACTIONS'
";
SQLQuery q = s.createSQLQuery( sql ).addEntity(
"t",AllTables.class );
```

- Workaround option 3: Query directly:

```
String sql = "select table_name , sysdate() from all_tables where
table_name = 'AUDIT_ACTIONS' ";
SQLQuery q = s.createSQLQuery( sql );
```

## HornetQ

<https://issues.jboss.org/browse/JBPAPP-10450>

If a HornetQ resource adapter (RA) is configured to connect to a remote JBoss Enterprise Application Platform/HornetQ server and the RA has `reconnect-attempts` set to `-1` (infinite) then the server hosting RA cannot be shut down cleanly if the remote server is not available.

<https://issues.jboss.org/browse/JBPAPP-6659>

The following examples provided with HornetQ don't work for various reasons:

- `jca-remote`
- `jms-bridge`
- `mdb-remote-failover-static`
- `mdb-remote-failover`
- `xarecovery`
- `embedded-simple`
- `embedded`
- `failover-manual-stop`
- `jaas`
- `spring-integration`
- `stomp-websockets`
- `stop-server-failover`
- `transaction-failover`

## Installer

<https://issues.jboss.org/browse/JBPAPP-10527>

While using the GUI installer, if the option to create shortcuts is unchecked and an automatic installation script is saved, the resulting installation script is invalid. The issue is that the shortcuts section of the installation script is empty and although this is a valid XML file, any attempt to use this script results in the following error. [ `Creating shortcuts`  
`java.lang.NullPointerException java.lang.NullPointerException at`  
`com.izforge.izpack.util.os.Unix_Shortcut.execPostAction(Unknown Source) at`  
`com.izforge.izpack.panels.ShortcutPanelAutomationHelper.runAutomated(Unknown`  
`Source) at`  
`com.izforge.izpack.installer.AutomatedInstaller.installPanel(Unknown`  
`Source) at`  
`com.izforge.izpack.installer.AutomatedInstaller.doInstall(Unknown Source)`  
`at com.izforge.izpack.installer.Installer.main(Unknown Source) [ Automated`  
`installation FAILED! ]`

<https://issues.jboss.org/browse/JBPAPP-10529>

While using the GUI installer, if the option to generate an automatic installation script is used, and a file of the specified name already exists, that file will be overwritten without warning. A workaround is to check if a file of the specified name already exists and if so, either rename it or specify a different file name.

<https://issues.jboss.org/browse/JBPAPP-10561>

The RPM installation method of JBoss Enterprise Application Platform does not include the JAR `jboss-common-beans.jar` in the CLASSPATH. As a result, launching an AppClient might fail with message:

```
DEPLOYMENTS IN ERROR:
  Deployment "ClientLauncherClassPath:0.0.0$MODULE" is
  in error due to: java.lang.RuntimeException: Error configuring property:
  roots for ClientLauncherClassPath:0.0.0$MODULE
```

To work around this issue, add `$JBOSS_HOME/lib/jboss-common-beans.jar` to the CLASSPATH. With this JAR added to the CLASSPATH, the above error message no longer occurs.

## Messaging

<https://issues.jboss.org/browse/JBPAPP-5124>

When using Sybase database with the JDBC driver Sybase jConnect JDBC driver v7 (Build 26502), a `MaxParams` attribute in the `sybase-persistence-service.xml` configuration file cannot be set higher than **481** due to a limitation of the `PreparedStatement` class in this driver. Unexpected failures can occur when the `MaxParams` attribute is set to a value higher than **481**. To avoid this issue, set `MaxParams` to a smaller value than **481**.



## CHAPTER 4. RESOLVED ISSUES

Resolved issues in JBoss Enterprise Application Platform 5.2.0 are listed by component.

### AOP

<https://issues.jboss.org/browse/JBPAPP-9198>

Previously, JBoss Pojo Cache's `pojocache-aop.xml` was missing a namespace on the `<aop>` tag. Consequently, if `jboss-cache-pojo.jar` was packaged in a location scanned by deployment scanners, the `JBossXBRuntimeException` was thrown. The namespace has been added to the `<aop>` tag and the `JBossXBRuntimeException` is no longer thrown in this scenario.

### Build

<https://issues.jboss.org/browse/JBPAPP-6752>

All licence files have been moved into one distribution RPM to avoid conflicts on operating systems that are not case-sensitive.

<https://issues.jboss.org/browse/JBPAPP-5302>

The file `properties-service.xml` was removed to minimise the number of unwanted services. Instead of using this file, the same result can be achieved by passing the properties in the server's startup.

### Clustering

<https://issues.jboss.org/browse/JBPAPP-9588>

JGroups `FD_SOCKET` protocol did not set the `TCP_KEEPALIVE` option on the server side of the socket. After a network interruption during which the client side had broken the connection, the server side sockets remained open until the JGroups channel was shut down. The `FD_SOCKET` protocol now sets the `TCP_KEEPALIVE` option on the server side of the socket. Now even if the client side of the socket is closed during a network interruption, the server side socket is closed by the `TCP_KEEPALIVE` timeout.

<https://issues.jboss.org/browse/JBPAPP-6447>

In previous versions of JBoss Enterprise Application Platform, an `UnmarshalException` (when a connection was reset) or an `EOFException` (when making a JNDI call as a server is shutting down) could occur. As a result of this the `UnmarshalException` was thrown to the caller with no attempt at failover. In this release of JBoss Enterprise Application Platform the `UnmarshalException` is caught and a new connection attempt is made to the supplied provider list. When possible the `UnmarshalException` now fails over transparently.

<https://issues.jboss.org/browse/JBPAPP-8152>

An issue that resulted in clients receiving errors has been resolved in this release. It was determined that HAJNDI was processing results before it was fully initialized, and the errors were coming from components that were not in the correct state. The issue has been corrected and HAJNDI now throws `javax.naming.ServiceUnavailableException`, which can be handled to retry the request automatically if a request comes in during start up.

<https://issues.jboss.org/browse/JBPAPP-8301>

In previous JBoss Enterprise Application Platform 5 releases a `NullPointerException` sometimes occurred when calling HAJNDI during startup. This was caused by `HARMIServerImpl` processes requests for HAJNDI before it is fully initialized. This would produce errors from components that were not in the correct state. As a resolution to this issue, `HARMIServerImpl` now throws `java.rmi.NoSuchObjectException` if a request comes in during startup, which causes an automatic retry in the client HAJNDI code.

<https://issues.jboss.org/browse/JBPAPP-8305>

It was found that, because HAJNDI continued to process requests while shutting down, clients could receive `CacheNotReadyException` errors from components that are not in the correct state. As a resolution to this issue, HAJNDI now throws `javax.naming.ServiceUnavailableException` if a request comes in during shutdown, and the shutdown process waits for existing requests to complete.

<https://issues.jboss.org/browse/JBPAPP-8302>

It was found that, because `HARMIServerImpl` continued to process requests for HAJNDI while shutting down, clients could receive `JBossLazyUnmarshallingException` errors from components that are not in the correct state. As a resolution to this issue, `HARMIServerImpl` now throws `java.rmi.NoSuchObjectException` if a request comes in during shutdown, and the shutdown process waits for existing requests to complete.

<https://issues.jboss.org/browse/JBPAPP-7904>

An issue that prevented a custom `HASingleton` election policy from being called if there was only one node in a cluster has been resolved in this release. The issue caused that a custom `HASingleton` election policy (such as a quorum policy) could not prevent a singleton from running in the cluster. In this release, the custom `HASingleton` election policy is always called and can therefore prevent a singleton from running in the cluster.

<https://issues.jboss.org/browse/JBPAPP-7418>

The JGroups `MPING` configurations ignored the `jboss.partition.udpGroup` system property (set with the `-u` option). Therefore, when setting the multicast address for clustering using `-u`, any channels using `MPING`, such as the JBoss Messaging data channel, used the default address instead of the configured address. The `jboss.partition.udpGroup` property has been added to the `MPING` configuration in the binding manager and setting the multicast address for clustering using `-u` now affects all clustering channels as expected.

<https://issues.jboss.org/browse/JBPAPP-9573>

The same name attributes were used on different tags in the POJO Cache's `pojocache-aop.xml` file. It resulted in an `IllegalStateException` error if `jbosscache-pojo.jar` was packaged in a location scanned by deployment scanners. To resolve this issue, the duplicate `name` attribute was removed.

<https://issues.jboss.org/browse/JBPAPP-9624>

When two threads attempted to create the same parent node at the same time, a race condition on locking occurred in JBoss Cache. As a result, one of the threads did not acquire a lock on the node, which could later lead to a deadlock. With this update, the threads now confirm that they created and have a write lock on the node and the deadlock no longer occurs.

<https://issues.jboss.org/browse/JBPAPP-7355>

Concurrent calls that started the same JBoss Cache instance caused errors. The calls are now synchronized so that concurrent access no longer occurs.

<https://issues.jboss.org/browse/JBPAPP-7850>

A database password was logged in plain text if a database connection was configured in a JBoss Cache loader and a connection error occurred. The log statement has been now altered so as not to log the password, which fixes this problem.

<https://issues.jboss.org/browse/JBPAPP-9220>

When using a JTS transaction manager, transaction commit could incorrectly share data between threads when under heavy load due to a race condition in JBoss Cache. This could result in lock time-out exceptions as locks were not freed correctly. With this update, the race condition no longer occurs, the transaction commit uses the correct data when using a JTS transaction manager under load, and transactions are committed correctly.

<https://issues.jboss.org/browse/JBPAPP-7500>

Inactivating a JBoss Cache region locked the entire subtree of the region's parent and never unlocked it. Consequently, the cache became unusable because of the locks. Inactivating a region now only locks the parent, not the entire subtree of the parent. As a result, the cache is no longer locked after inactivation of a region.

<https://issues.jboss.org/browse/JBPAPP-9560>

When an exception occurred during marshalling in `CommandAwareRpcDispatcher`, the exception was logged without its context, making diagnostics difficult. The message has now been modified so that it clearly states that it occurred during marshalling.

<https://issues.jboss.org/browse/JBPAPP-9574>

JBoss Cache unmarshalled return values that were going to be discarded during replication. When the cache contained objects in an isolated classloader, it was unable to unmarshall the objects and an exception was thrown, preventing the replication from succeeding. The return values are now no longer unmarshalled since they will be discarded. The exception is therefore not thrown and the replication finishes successfully.

<https://issues.jboss.org/browse/JBPAPP-7709>

Due to issues in the Oracle driver, the `JDBCCacheLoader` did not work with an Oracle database if the serialized form of the data exceeded a certain size (a couple of kB). The `cache.jdbc.node.useSetBlob` cache loader flag has been added to solve this problem: when the flag is set, JDBC uses a different JDBC call of the Oracle driver implementation and the `JDBCCacheLoader` works as expected with Oracle databases under these circumstances.

<https://issues.jboss.org/browse/JBPAPP-9604>

JBoss Cache used the `useOutOfBandMessage` message hard-coded to `false` when making remote procedure calls. Consequently, RPC messages that were meant to be processed immediately, such as a transaction commit, had to wait for other messages to be processed first, which could cause deadlocks. With this update, the `useOutOfBandMessage` is no longer set to `false`. Out-of-band RPC messages are now processed immediately and can no longer cause deadlocks.

<https://issues.jboss.org/browse/JBPAPP-9639>

JBoss Cache transaction `ROLLBACK` messages that were processed before the corresponding `PREPARE` were ignored. This caused the `PREPARE` to acquire locks and release them only after the

transaction timed out. JBoss Cache now keeps track of **ROLLBACK** messages processed before the corresponding **PREPARE** message, and aborts **PREPARE** if the corresponding **ROLLBACK** was already processed.

<https://issues.jboss.org/browse/JBPAPP-9344>

When requesting missing messages to be retransmitted, a node sometimes requested the latest received message instead of the missing one. If it had not previously received any messages from the member in these cases, it requested message 0 (which is invalid). This behaviour caused unnecessary retransmission requests and the following warning messages were logged:

```
WARN [org.jgroups.protocols.pbcast.NAKACK] (00B-19,10.1.1.1:55200)
(requester=10.1.1.2:55200, local_addr=10.1.1.1:55200) message
10.1.1.1:55200::0 not found in retransmission table of 10.1.1.1:55200:
[0 : 7 (7) (size=7, missing=0, highest stability=0)]
```

With this update, JGroups do not request retransmission of message 0 or a message it already has received. As a result, no redundant retransmission requests are sent and the above mentioned warning messages are not logged.

<https://issues.jboss.org/browse/JBPAPP-7472>

A race condition in JGroups could lead to a **ConcurrentModificationException** in the **BasicConnectionTable.retainAll()** call. The race condition occurred on the members variable when using the TCP protocol. The appropriate synchronization has been implemented to prevent concurrent access to the variable so that the **ConcurrentModificationException** can now no longer occur.

<https://issues.jboss.org/browse/JBPAPP-7456>

If a protocol considers a cluster node dead, the protocol sends a **SUSPECT** message. The **VERIFY\_PROTOCOL** then pings the node to verify that the node is unresponsive. If the node does not respond before the **VERIFY\_SUSPECT** time-out period elapses, the node is kicked from the cluster. Previously, after a cluster member was kicked and then rejoined the cluster, the member could be kicked again if it joined the cluster before the time-out period had elapsed. This happened because the member was not removed from the list of the suspected members after it has left the cluster. Now, a node that leaves the cluster is removed from the list of suspected members so that the member is not kicked after rejoining the cluster again, even if it rejoins the cluster before the timer expires.

## Consoles

<https://issues.jboss.org/browse/JBPAPP-5925>

When using the IBM JDK and logging onto the web console for the first time, a **java.lang.UnsupportedOperationException** error occurred. The root cause of this issue was IBM's Java implementation. The problem has been resolved by working around the issue with IBM's JDK so that the error no longer occurs when logging in to the web console.

## Documentation

<https://issues.jboss.org/browse/JBPAPP-9433>

The *Installation Guide* stated that the GUI installer configures JBoss Enterprise Application Platform to run as a service on Red Hat Enterprise Linux. However, this is only the case when the product is installed from an RPM package. With this update, the text has been corrected to reflect this fact.

<https://issues.jboss.org/browse/JBPAPP-5491>

A documentation oversight lead to SNMP-related content not being included in the JBoss Enterprise Application Platform 5 documentation suite (it was present in the 4.3 suite). The information has now been included in the latest release of the *Administration and Configuration Guide*.

<https://issues.jboss.org/browse/JBPAPP-9635>

The *HTTP Connectors Load Balancing Guidedid* did not discuss load providers and how to control the metrics that are reported to the http-side module. The information has been added to the guide and is available in the Load Metrics chapter.

<https://issues.jboss.org/browse/JBPAPP-9804>

The *Administration and Configuration Guidedid* did not document that datasource properties `<background-validation>` and `<background-validation-minutes>` have been deprecated. Notes on deprecation have been added and the property descriptions have been modified respectively.

<https://issues.jboss.org/browse/JBPAPP-10294>

A procedure for switching to a production database using the new Database Configuration Tool has been added to the *Administration and Configuration Guide*

<https://issues.jboss.org/browse/JBPAPP-4550>

Previously, the *Installation Guide* provided incorrect information on how to define the `SuckerPassword`, which is used by nodes during messaging for authentication in clustered environment. With this update, the correct information has been added to the *Messaging Guide* and the original information has been removed from the *Installation Guide*.

<https://issues.jboss.org/browse/JBPAPP-5499>

In response to a request, details about how to install `mod_cluster`, `mod_cluster support` and `mod_jk` in JBoss Enterprise Application Platform 5 have been added to the *Installation Guide*.

<https://issues.jboss.org/browse/JBPAPP-3207>

Customers were experiencing difficulties running the non-clustered Messaging examples using the `All` and `Production` server profiles because the *Messaging User Guide* did not contain clear instructions. Most Messaging examples are designed to run on non-clustered server profiles (all profiles, except `All` and `Production`). The chapter has been improved by separating the examples into clustered and unclustered sets, and also instructs the user to review the `readme` file shipped with each example for configuration instructions.

<https://issues.jboss.org/browse/JBPAPP-7146>

The *Security Guide* did not include the information that user names and roles are case sensitive. A note reflecting this fact has been added.

<https://issues.jboss.org/browse/JBPAPP-9595>

The *Performance Tuning* chapter was removed from the *Administration and Configuration Guide* and is now substituted by the *Performance Tuning Guide*.

<https://issues.jboss.org/browse/JBPAPP-9878>

Previously, the *HornetQ User Guide* did not contain information on HornetQ HA supported shared stores. The appropriate information has been added.

## EJB

<https://issues.jboss.org/browse/JBPAPP-6624>

A memory leak was discovered in the `InfinitePool`'s implementation of the `ThreadLocalPool` which could eventually lead to `OutOfMemory` exceptions. The `InfinitePool` maintained a list of active `beanContexts` but did not override the `discard` method so a small memory leak occurred every time this method was called. A `discard` method was implemented, preventing memory leaks occurring.

<https://issues.jboss.org/browse/JBPAPP-8123>

If a class did not implement `EJBObject` but was annotated as `@Stateful`, JBoss incorrectly considered the respective object not to be an EJB. This happened because the code failed to handle the returned EJB-annotation results. With this update, the EJB annotation is handled correctly and the object is identified as an EJB Bean as expected.

<https://issues.jboss.org/browse/JBPAPP-7685>

Previously, the `timeout()` methods on an EJB3 timer bean ignored the `@RunAs` annotation due to a missing interceptor reference in the XML defining EJB3 interceptors. With this update, the `RunAs` interceptor reference has been added and the `timeout()` method is run with the matching role as expected.

<https://issues.jboss.org/browse/JBPAPP-7724>

If a Stateful Session Bean reached its `removalTimeout`, the system removed the bean without executing the `@PreDestroy` lifecycle. JBoss will now execute the `@PreDestroy` annotated method of the Stateful Session Bean before removing the bean.

<https://issues.jboss.org/browse/JBPAPP-8035>

Extensive use of the `ejbTimeout()` callbacks in an EJB resulted in "*failure to create native thread*" OOM (out of memory) errors rendering the server unusable. Now, each server has a `ScheduledExecutorService`, which registers any deployed EJB. The service has a restricted pool size of threads which process the timeout callbacks. The pool size is set to 50 threads by default.

<https://issues.jboss.org/browse/JBPAPP-7292>

In previous versions of JBoss Enterprise Application Platform 5, it was found that the EJB3 deployer could slow a large deployment down significantly. This was because, during a deployment, the deployer scanned the entire archive to locate client deployment descriptors. It would ignore the deployment of any it found. This issue has been resolved by ensuring that all deployment descriptors are located directly in the `META-INF` directory, giving the deployer a single location to scan and leading to faster deployments.

<https://issues.jboss.org/browse/JBPAPP-4428>

When the `WebServiceContext` was injected into an EJB3 interceptor the following error was reported:

■ Caused by: `java.lang.ArrayStoreException`:

```

org.jboss.injection.WebServiceContextPropertyInjector
at java.util.AbstractCollection.toArray(AbstractCollection.java:171)
at org.jboss.ejb3.interceptor.InterceptorInjector.<init>
(InterceptorInjector.java:42)
at org.jboss.ejb3.EJBContainer.processMetadata(EJBContainer.java:620)
at
org.jboss.ejb3.Ejb3Deployment.processEJBContainerMetadata(Ejb3Deployment
.java:434)
at org.jboss.ejb3.Ejb3Deployment.start(Ejb3Deployment.java:543)
...

```

The cause of this issue was that the `WebServiceContextPropertyInjector` did not extend the `PojoInjector`. This has been corrected and this error no longer occurs.

#### <https://issues.jboss.org/browse/JBPAPP-923>

Propagation of an Extended Persistence Context (XPC) was not taking into account the existence of a transaction, with the XPC always being propagated. The handling of XPC has been modified so that if there is no transaction active the XPC's propagation is ignored so that the bean being invoked has its own Persistence Context instead of the XPC. This behavior can be overridden by adding the following to `run.conf`:

```

{{JAVA_OPTS="$JAVA_OPTS -DJBPAPP-923.alwaysPropagate=true"}}

```

#### <https://issues.jboss.org/browse/JBPAPP-7488>

Persistence units that had exploded packages in a `<jar-file>` tag did not have their entities scanned. It returned the `vfswfile` protocol for exploded packages declared in `<jar-file>` which were not scanned by Hibernate `EntityManager`. The class `JarVisitorFactory` returned an incorrect implementation of `AbstractJarVisitor` for `vfswfile` (returning `InputStreamZippedJarVisitor`). This issue has been resolved by modifying `JarVisitorFactory` so that it returns the correct implementation and as a result, `ExplodedJarVisitor` is able to scan entities in exploded packages appropriately.

#### <https://issues.jboss.org/browse/JBPAPP-9341>

Because unlinked collections were used by the EJB-QL compiler to store join paths of processed queries, the order of individual parts of produced SQL queries was different with different versions of the Java Development Kit. While the queries produced identical results, the differences affected execution plan performance on some database systems. With this update, the unlinked collections have been replaced with linked ones. As a result, the produced SQL queries are now identical with different Java Development Kit versions and execution plan performance is no longer affected.

#### Embedded Jopr

#### <https://issues.jboss.org/browse/JBPAPP-8084>

When using the administration console to update an application (EAR or WAR) deployed to the `farm` directory, the target application was removed from the `farm` directory and deployed to the `deploy` directory. This resulted result in the `farmed` attribute value being `null` when the administration console UI expected it to be set to `true`. This issue has been resolved by ensuring that existing applications that are marked as `farmed` continue to be deployed in the `farm` directory, which ensures `farmed=true` on the updated deployment.

#### <https://issues.jboss.org/browse/JBPAPP-8222>

When an XML parser in strict mode parsed a deployment descriptor found in `admin-console.war`, a failure occurred due to a missing version attribute in the `persistence.xml` descriptor. This issue has been resolved by adding the missing version attribute to the `persistence.xml` file for `rhq-core-domain.jar`.

<https://issues.jboss.org/browse/JBPAPP-7551>

When using Microsoft Internet Explorer (IE) 7 or IE 8 in IE 7 compatibility mode, the administration console's main panel was sometimes blank. This was due to a Microsoft CSS extension used by the administration console that had been deprecated in IE 6 and removed in IE 7. This issue has been resolved by updating the administration console's CSS to use a standard style rendering that is compatible with IE 6 and later.

<https://issues.jboss.org/browse/JBPAPP-9459>

When creating a data source in the Admin Console, the data source mapping type did not provide the `MySQL` option. The data source type mapping property has been changed to a text field and the `MySQL` mapping type can now be entered as the data source mapping type.

<https://issues.jboss.org/browse/JBPAPP-9461>

On access, the Admin Console turned off global caching of resources contained within JAR files. This resulted in serious performance issues. With this update, the console explicitly re-enables default caching on `URLConnection` after it is accessed and the performance problems no longer occur in this scenario.

## Hibernate

<https://issues.jboss.org/browse/JBPAPP-6395>

When the Criteria API was used with `LEFT OUTER JOIN` to add criteria to children, the child collections contained only children matching the criteria. This behavior also applied when no Filters were used. For example:

```
criteria.createCriteria("children", JoinFragment.LEFT_OUTER_JOIN)
```

The root cause of this issue was that the criteria were not being properly propagated. This has now been resolved and these criteria are now properly applied.

<https://issues.jboss.org/browse/JBPAPP-6013>

In the server's logs, Hibernate Annotations' version number was being reported as `3.4.0.GA_CP01` and Hibernate EntityManager's was reported as `WORKING` when both were in fact the same version: `3.4.0.GA_CP01`. This has now been fixed and the correct version numbers of both components are recorded in the server's logs.

<https://issues.jboss.org/browse/JBPAPP-6906>

When the `EXISTS` statement was negated in Hibernate HQL, it was incorrectly translated to SQL without the negation. This issue has been resolved and the statement is now correctly translated.

<https://issues.jboss.org/browse/JBPAPP-8370>

With `JBossCache` configured as a second-level cache for Hibernate, and `TimeStampsRegionImpl` enabled, a Null Pointer Exception (NPE) could occur on initialization of the cache. To prevent this occurring, an additional check has been added for these circumstances, preventing the NPE and ensuring the cache is initialized correctly.



<https://issues.jboss.org/browse/JBPAPP-7998>

It was found that when a non-English language is used in an Oracle database, constraint violation could not be correctly extracted and reported. This was because in the Oracle SQL dialect, constraint violation name extraction was English-specific. A patch has been applied to make constraint violation name extraction language-neutral. Now, when Oracle is using a non-English language, the constraint can be seen.

<https://issues.jboss.org/browse/JBPAPP-8682>

A bug was identified in the `SchemaUpdate` function where, although it read the foreign keys metadata from the database, it ignored `_default_schema_` and `_default_catalog_` settings, resulting in incorrect SQL statements. This issue has now been fixed and these default settings are honored.

<https://issues.jboss.org/browse/JBPAPP-9227>

Hibernate returned an exception stating that the primary key of a child entity was ambiguous when updating the primary key of the child element while both entities, the entity of the parent element and the entity of the child element, were mapped with joined subclass method. This happened because Hibernate generated an incorrect SQL leaving out the table alias from the `id` column. An upstream fix that fixes this issue has been backported to Hibernate and Hibernate now generates the correct SQL under these circumstances.

<https://issues.jboss.org/browse/JBPAPP-7072>

If Hibernate's `EntityManager` was used in a CMT Bean and a `GenericJDBCException` occurred, Hibernate would initiate a rollback even if there was no active transaction, resulting in the exception:

```
<org.hibernate.ejb.AbstractEntityManagerImpl> Unable to mark for
rollback on PersistenceException:
java.lang.IllegalStateException:
[com.arjuna.ats.internal.jta.transaction.arjunacore.nosuchtx]
[com.arjuna.ats.internal.jta.transaction.arjunacore.nosuchtx] No such
transaction!
```

This issue has been resolved so a rollback is no longer initiated in these circumstances.

<https://issues.jboss.org/browse/JBPAPP-7291>

When executing a SQL query that returns values of the `char` SQL data type, a value of the `java.lang.Character` data type was returned, regardless of the `char` data type length. This caused that only the first character of the value was returned. A value of the `java.lang.String` data type is now returned when the length of the `char` type is larger than 1.

## HornetQ

<https://issues.jboss.org/browse/JBPAPP-6659>

The HornetQ examples were dependent on a build script - `build.sh`, which was not included with the examples. The dependency has now been removed so the only dependency for installing the scripts is the `ant` utility. To build the examples, perform these steps:

1. Open a terminal;
2. Navigate to the example directory root containing the `build.xml` file;

### 3. Execute `ant` to build and deploy the examples.

<https://issues.jboss.org/browse/JBPAPP-8561>

An issue was discovered in which when `destroy()` was called on a managed connection, `stop()` was called on the managed connection factory which stopped the `ServerLocator` and subsequently all sessions all the managed connections associated with that factory. The root cause of this issue has now been resolved.

<https://issues.jboss.org/browse/JBPAPP-8786>

A bug was discovered which caused a node in HornetQ clusters using collocated back up servers to become unresponsive the shutdown signal was sent. This was caused by the JBoss Shutdown Hook thread being blocked from joining a `JMSServerManagerImpl` thread, because it, in turn, was waiting for a lock that was already held by the JBoss Shutdown Hook thread. This issue was corrected by changing the synchronization semantics of the related methods and by changing the order of operations. HornetQ cluster nodes now shut down gracefully when collocated.

<https://issues.jboss.org/browse/JBPAPP-8168>

If a message consumer crashed, any messages which it had consumed but not acknowledged did not have their delivery count updated. This issue has now been fixed so that delivery counts are accurate.

<https://issues.jboss.org/browse/JBPAPP-8151>

If a message was paged and the session which sent the message was rolled back, the paged message was not removed until the server was restarted. The cause of this issue has now been fixed and paged messages are correctly removed.

<https://issues.jboss.org/browse/JBPAPP-8434>

In certain circumstances messages were lost when using a message selector. The root cause was identified and fixed so that message references are maintained.

<https://issues.jboss.org/browse/JBPAPP-8578>

It was found that a `discovery-group` would effectively ignore the `group-address` and simply bind to `0.0.0.0`. This could potentially lead to cluster "cross-talk" where applications listening on a `MulticastSocket` receive packets associated with a different multicast address. The problem was caused by EAP using the wrong constructor for `java.net.MulticastSocket`. A patch has ensured EAP uses the correct constructor and adds a log message to warn the user if the expected configuration failed.

<https://issues.jboss.org/browse/JBPAPP-7652>

An issue was found where JNDI bindings added using `JConsole` were valid only until the instance was restarted, when all changes would be lost. This has been resolved and all such changes are persistent.

<https://issues.jboss.org/browse/JBPAPP-8731>

In circumstances where a JMS server backup is started and then the backup server is shutdown before the backup goes live, an `IllegalStateException` can be thrown (on the backup server). This is because any calls to create new connection factories and destinations for the backup server are cached and, if the server is shut down, EAP will attempt to undeploy them only to find they do not exist. This issue has been corrected with a patch that caches the relevant shutdown commands (just as is done with the start commands) so that they do not get executed until the server is active.

<https://issues.jboss.org/browse/JBPAPP-8769>

A bug was found wherein sending extremely large messages had the potential to lock the queue while they were being transmitted. This could, in some extreme cases, lead to timeouts. This bug was fixed in a cumulative patch.

<https://issues.jboss.org/browse/JBPAPP-8997>

When refreshing the metrics page of a HornetQ message queue, a message indicating that there were no numeric metrics available appeared occasionally. This was caused by a race condition identified in [JBPAPP-9406](#). The race condition has been corrected to prevent the management view from becoming out-of-sync and the problem no longer occurs.

<https://issues.jboss.org/browse/JBPAPP-9002>

In previous releases of EAP, if `BroadcastGroupImpl` was unable to broadcast, it logged an error every 5 seconds. This release changes that behavior so that the error is only logged once. A message is also logged when the condition changes.

<https://issues.jboss.org/browse/JBPAPP-9027>

If a `javax.jms.ObjectMessage` with a `java.lang.reflect.Proxy` payload was sent to a message queue, the `getObject()` call on the message failed with a `JMSEException`. This happened because the underlying `org.hornetq.utils.ObjectInputStreamWithClassLoader` class did not implement the `java.io.ObjectInputStream.resolveProxyClass()` method. This update provides a suitable implementation of the method.

<https://issues.jboss.org/browse/JBPAPP-9258>

An issue was found wherein JMX Topic attributes were not available in a cluster. The issue was caused by a `NullPointerException` in the code which looped through the subscriptions to gather metrics and caused all clustered nodes to incorrectly report topic attributes as `0`. This issue was fixed by checking for null on the relevant variable.

<https://issues.jboss.org/browse/JBPAPP-9393>

In previous versions of JBoss Enterprise Application Platform 5, when a HornetQ "connector" was bound to `0.0.0.0` HornetQ would tell a remote client to connect to the same address. However, `0.0.0.0` is meaningless to a remote client. This issue was resolved by detecting the `0.0.0.0` binding and replacing that value with the result of `InetAddress.getLocalHost().getHostName()` so that the remote client can connect properly.

<https://issues.jboss.org/browse/JBPAPP-9617>

In HornetQ, when a message was imported from an XML file, the message body was null due to a `NullPointerException` that occurred when acquiring message attributes. The process of acquiring message attributes has been fixed and the problem no longer occurs.

<https://issues.jboss.org/browse/JBPAPP-9616>

It was reported that the HornetQ journal importer would throw a `NullPointerException` when passed an argument of `false` for the transactional command line parameter. This was found to be occurring because the XML importer uses two different HornetQ sessions; one for normal send operations and one for management operations. In certain circumstances the management session was not instantiated properly and, when it was used, the exception was thrown. The XML importer has been updated and the issue no longer presents.

<https://issues.jboss.org/browse/JBPAPP-9960>

An issue was discovered which affected HornetQ high-availability clusters and failover. If, in the `ra.xml` file, the hostname was specified as a string, any active session would not failover to a backup node. The root cause of this issue has been resolved and hostnames in string format are no longer a problem for failover.

<https://issues.jboss.org/browse/JBPAPP-8067>

An issue was identified with HornetQ where paged messages would not be expired if there was no active consumer. The expiry scan was enhanced so that it now includes paged messages in its analysis.

<https://issues.jboss.org/browse/JBPAPP-9669>

When the HornetQ installer was used to install a backup server using the `-Dbackup=true` parameter, the following message would result:

```
15:18:09,615 ERROR [ProfileServiceBootstrap] Failed to load profile:
Summary of incomplete deployments (SEE PREVIOUS ERRORS FOR DETAILS):

DEPLOYMENTS MISSING DEPENDENCIES:
  Deployment "AS5RecoveryRegistry" is missing the following
dependencies:
  Dependency "TransactionManager" (should be in state "Installed", but
is actually in state "*** NOT FOUND Depends on 'TransactionManager' **")

DEPLOYMENTS IN ERROR:
  Deployment "TransactionManager" is in error due to the following
reason(s): ** NOT FOUND Depends on 'TransactionManager' **
```

The cause of this issue was that in deploying the backup server, the deployment of the AS5RecoveryRegistry mbean was also attempted but failed because of missing dependencies. To resolve this issue the AS5RecoveryRegistry mbean has been excluded from deployment.

<https://issues.jboss.org/browse/JBPAPP-8377>

An issue was identified in which it was possible to have duplicate resource recovery entries for one recovery manager, resulting in recovery scans conflicting with each other. This problem has been resolved by implementing auto-discovery of resources instead of manual configuration.

<https://issues.jboss.org/browse/JBPAPP-9603>

A memory leak was identified in the paging module. When a durable subscription was removed, the paging system was not notified, and it would keep paging for that nonexistent subscription. As a result, the resources used to maintain paging would increase and only a restart of JBoss Enterprise Application Platform would release them. The handling of durable subscriptions has been changed so that the paging system is notified, resolving the memory leak.

## Installer

<https://issues.jboss.org/browse/JBPAPP-7674>

Previous versions of the JBoss Enterprise Application Platform 5 graphical installer required `root` privileges to run. Consequently, all files created or modified by EAP (including the launcher icons) inherited these permissions and could not be used by other accounts. To correct this behavior, the

requirement for the graphical installer to be run as *root* has been removed (the documentation has also been updated) and unprivileged users can now install and run an EAP 5 instance using the graphical installer.

<https://issues.jboss.org/browse/JBPAPP-9665>

The JTA RPM was missing a symbolic link and the RPM installation process returned the following warning: `primary link for jta_api must be /usr/share/java/jta_1_0_1B_api.jar`. The symbolic link has been added to the RPM package and the warning is no longer displayed on RPM installation.

<https://issues.jboss.org/browse/JBPAPP-7638>

The "Hotel" demonstration application would not run on Microsoft Windows Server because of restrictions implemented via User Access Control (UAC). The batch file used to start the application has been modified so that it complies with UAC's restrictions and so it now starts correctly.

<https://issues.jboss.org/browse/JBPAPP-8115>

When the installer was launched from the command line and WSNative was chosen as the web services stack, the following error message would occur when all prompts were completed. The root cause of this issue was a missing file. This file has been added and the installer now works correctly.

```
[ ERROR: The following error occurred while executing this line:
.../jboss-as/scripts/install-cxf.xml:8: Basedir .../jboss-as/jbossws-
cxf-installer does not exist ]
com.izforge.izpack.installer.InstallerException: The following error
occurred while executing this line:
.../jboss-as/scripts/install-cxf.xml:8: Basedir .../jboss-as/jbossws-
cxf-installer does not exist
```

## JCA

<https://issues.jboss.org/browse/JBPAPP-7786>

The JBossTS transaction recovery manager was previously improved to prevent a stale JDBC connection for XA recovery by checking the validity of the connection before making any calls to the driver. In the case of the DB2 JDBC driver however, the validity check caused a connection to the database to be opened and remain open, preventing a planned shutdown of the DB2 RDBMS instance even though no activity was in progress. To prevent this occurring, a new system property - `recover-connection-validation` - has been created, allowing validation to be turned off, for example: `-Drecover-connection-validation=false`.

<https://issues.jboss.org/browse/JBPAPP-7863>

When a thread's interrupted status was `true` and it called `java.sql.DataSource.getConnection()` for a JCA datasource, it resulted in error message:

```
javax.resource.ResourceException: Interrupted while requesting permit!
Waited 0 ms
```

To assist debugging the following text has been added to this error message:

The 0 ms wait time suggests the thread was already interrupted before attempting to acquire the connection. Ensure the calling thread's interrupted status is cleared before calling `getConnection()`.

<https://issues.jboss.org/browse/JBPAPP-9236>

When a security domain was set in the data source configuration, a race condition between the `IdleRemover` thread and the `JCA PoolFiller` thread caused a connection leak, when a connection was created outside of the connection pool. Consequently, the pool failed to contain the required minimum number of connections. The race condition has been fixed and the pool connections are created correctly.

<https://issues.jboss.org/browse/JBPAPP-9237>

If an application called the `setQueryTimeout()` method on a statement object, the database query time-out did not become effective. This happened when query-timeout was set in the data-source configuration. With this update, the time-out query is overwritten by the application that called the `setQueryTimeout()` method as expected.

<https://issues.jboss.org/browse/JBPAPP-7464>

When the `org.jboss.resource.adapter.jdbc.xa.XAManagedConnectionFactory.equals()` method was executed simultaneously from multiple threads, a JCA-layer deadlock occurred. This update synchronizes three methods used within the `equals()` method, and changes the object type of the objects that were accessed on the deadlock from `xaProps` to `HashMap`. As a result, the `equals()` method can now be executed simultaneously by multiple threads and no deadlock occurs.

<https://issues.jboss.org/browse/JBPAPP-9384>

The `<url-selector-strategy-class-name>` property in a local-tx-datasource definition was ignored and the default implementation was used. This happened because the datasource deployer was setting the `UrlSelectorStrategyClassName` instead of `URLSelectorStrategyClassName` on the class `BaseWrapperManagedConnectionFactory.java`. This mistake has been fixed and the property is picked up and applied as expected.

<https://issues.jboss.org/browse/JBPAPP-9463>

Previously, when using JDBC to connect to XA datasources, it was not possible to use a custom URL selector. To allow to do so, the `XAData` class was made public.

<https://issues.jboss.org/browse/JBPAPP-9470>

The `org.jboss.resource.adapter.jdbc.remote WrapperDataSourceService.doStatementMethod` method did not return a proxied connection. Consequently, when the `statement.getConnection()` JDBC API method was called from an application, the following exception was thrown:

```
Caused by: java.lang.IllegalAccessException: Method=public abstract
java.sql.Connection java.sql.Statement.getConnection() throws
java.sql.SQLException does not return Serializable
at
org.jboss.resource.adapter.jdbc.remote.WrapperDataSourceService.doStatem
entMethod(WrapperDataSourceService.java:422)
```

```

at
org.jboss.resource.adapter.jdbc.remote WrapperDataSourceService.invoke(WrapperDataSourceService.java:227)
at sun.reflect.GeneratedMethodAccessor165.invoke(Unknown Source)
at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)

```

With this update, a proxied connection is returned, which eliminates the above mentioned exception and enables successful calls of the `statement.getConnection()` method.

<https://issues.jboss.org/browse/JBPAPP-7543>

In the event the Oracle database driver reported an `SQLException` error, this was not treated in the same way as other error events. The handling of such an error has been changed so that it is treated as a fatal database connection error. Oracle database connections are now handled using `oracle.jdbc.OracleConnection` instead of the deprecated `oracle.jdbc.driver.OracleConnection` method. The method of checking the status of an Oracle database connection has been changed, using `pingDatabase()` instead of the deprecated `pingDatabase(int)`.

## JMX

<https://issues.jboss.org/browse/JBPAPP-7295>

The `traceInstructions`, `traceMethodCalls`, `runGarbageCollector` and `runFinalization` methods in `org.jboss.system.server.jmx.JMXKernel` were empty. Calling these methods was therefore successful, but no operation was actually performed by them. The code of the methods has been extended to call the alike named methods in `java.lang.Runtime`. The methods are now no longer empty and the expected operations are performed when the methods are called.

## Microcontainer and Deployers

<https://issues.jboss.org/browse/JBPAPP-9468>

When the `jboss.vfs.forceCanonical` property was set to `true`, VFS did not correctly cache `VFSXXXContext` objects for paths that included symbolic links. With this update, all symlinks are resolved to a path within the permanent root directory specified in the `VFSCache` bean defined in the `$JBOSS_HOME/server/$CONFIG/conf/bootstrap/vfs.xml` file, ensuring that the `VFSXXXContext` objects are now cached correctly.

## mod\_cluster

<https://issues.jboss.org/browse/JBPAPP-8956>

Due to a change in the `proxy_balancer` structure between 2.2.15 and 2.2.17 (which is the version used to compile EAP), a `Segmentation Fault` was thrown from `mod_proxy_cluster`. so when running more than one EAP server cluster behind a single Apache server. This was because all the clusters used the same domain name (usually the cluster name). This issue was resolved by reading the size of the `proxy_balancer` structure for the `httpd_mod_proxy` tables and using that size in `mod_cluster` logic. The `Segmentation Faults` no longer present.

<https://issues.jboss.org/browse/JBPAPP-9396>

When using JBoss HTTP Connector (mod\_cluster), an RPC (remote procedure call) failure could cause deployment of a web application to fail because RPC call exceptions were propagated to the web container. Such exceptions are no longer propagated and the problem no longer occurs.

## Naming

<https://issues.jboss.org/browse/JBPAPP-8838>

Naming lookups have been improved so that if a `ServiceUnavailableException` occurs when a `context.lookup` call is made, the call is retried.

## PicketLink

<https://issues.jboss.org/browse/JBPAPP-7853>

The `SAMLSubjectParser` did not support the `<saml2:NameID>` subelement within the `<saml2:SubjectConfirmation>` subelement of the `<saml2:Subject>` element. If it found the `<saml2:NameID>` subelement, it parsed it incorrectly and threw the following exception:

```
java.lang.ClassCastException: com.ctc.wstx.evt.CompactStartElement
cannot be cast to javax.xml.stream.events.EndElement
    at
org.picketlink.identity.federation.core.parsers.saml.SAMLSubjectParser.parse(SAMLSubjectParser.java:123)
    at
org.picketlink.identity.federation.core.parsers.saml.SAMLAssertionParser.parse(SAMLAssertionParser.java:148)
    at
org.picketlink.identity.federation.core.parsers.saml.SAMLParser.parse(SAMLParser.java:76)
```

This update modifies the `SAMLSubjectParser` to support the `<saml2:NameID>` subelement within the previously mentioned elements. The `<saml2:NameID>` subelement is now parsed correctly and the exception cited above is no longer thrown.

## Profile Service

<https://issues.jboss.org/browse/JBPAPP-9213>

When the user attempted to create a JMS queue or topic in the admin-console, the process failed due to missing default values of the `expiryQueue`, `DLQ`, and `serverPeer` properties. Now, the properties are populated with acceptable default values within the UI in the admin-console when a new message queue or topic is being created.

<https://issues.jboss.org/browse/JBPAPP-9406>

Due to a race condition when clearing and reloading the ProfileService ManagementView, the Admin Console and JBoss ON randomly report that managed components were missing or not available. The respective reloading process has been fixed and the race condition no longer occurs.

## Remoting

<https://issues.jboss.org/browse/JBPAPP-8912>



If you had configured JBossWS clients to use the socket factory returned by `HttpsURLConnection.getDefaultSSLSocketFactory()`, then you would receive an ERROR log message even though the configuration would work properly. The code causing this issue has been cleaned up, and a new property is available to tell JBossWS to use the default SSL socket factory: `StubExt.PROPERTY_DEFAULT_SSL_SOCKET_FACTORY`.

## Scripts and Commands

### <https://issues.jboss.org/browse/JBPAPP-7436>

Due to missing quotation of certain echo statements in the `run.bat` script, JBoss Enterprise Application Platform failed to start if the `JBOSS_HOME` path contained one or more parentheses (e.g. `C:\JB-EAP(test)\jboss-eap-5.1\jboss-as`), and returned the following error message:

```
\Jboss-5.1.0\bin\run.conf.bat was unexpected at this time.
```

Double quotes have been added around the echo statements in the `run.bat` file, ensuring that JBoss Enterprise Application Platform starts successfully even if the `JBOSS_HOME` path contains parentheses.

### <https://issues.jboss.org/browse/JBPAPP-9581>

JBoss Enterprise Application Platform attempted to load a per-profile `run.conf` file from the default location in `$JBOSS_HOME/server/$PROFILE`. Consequently, per-profile `run.conf` files stored in a different location were not loaded. The `--run-conf=...` option (on Windows: `-r`) has been added to the `run.sh` script, allowing to specify the location of the `run.conf` file that should be used. Using this option, per-profile `run.conf` files stored in another than the default location can now be loaded correctly.

### <https://issues.jboss.org/browse/JBPAPP-7514>

In JBoss Enterprise Application Platform version 5.1.1, the `run.sh` startup script was enhanced to include a test for a 64-bit JVM. However, this test was not working on Solaris because the standard `grep` command does not support the `-e`, which specifies the regular expression to be used in the search. This issue was resolved by explicitly using `/usr/xpg4/bin/grep` since this does support the regular expression parameter `-e`.

### <https://issues.jboss.org/browse/JBPAPP-8148>

An issue was found with the the JBoss Native service batch file on Windows Server. When JBoss Enterprise Application Platform was started, the service batch file (`service.bat`) searched for its associated configuration batch file in the wrong directory, resulting in the error message `Config file not found $JBOSS_DIST/native/sbin/run.conf.bat`. The cause of the issue was that it was looking for the configuration batch file in `$JBOSS_HOME/bin` when instead it was in `$JBOSS_DIST/native/sbin`. The service batch file has now been corrected with the correct path specified.

## Seam/Seam2

### <https://issues.jboss.org/browse/JBPAPP-6376>

The DVD Store demonstration Seam application provided with JBoss Enterprise Application Platform failed to deploy. The application is now updated and will successfully deploy.

<https://issues.jboss.org/browse/JBPAPP-7269>

When using the `<s:token />` tag in JSF, the `javax.faces.ClientToken` cookie that is created sometimes contained illegal characters, for example: semicolon ";", resulting in the following error message:

```
org.jboss.seam.ui.UnauthorizedCommandException: viewId:  
/restricted/desktop.xhtml - Form signature invalid
```

The cause of this issue was that the code which created the cookie's value did no validation. The code has been amended so that the cookie can contain only letters and digits, avoiding the circumstances leading to the error message above.

<https://issues.jboss.org/browse/JBPAPP-8041>

The execution of the "ant eclipseclasspath" task failed as the OpenId dependency pom file was incorrectly included in the generation of Eclipse project files. The ant task now ignores the pom files and the problem no longer occurs.

<https://issues.jboss.org/browse/JBPAPP-8322>

As JNDI transaction lookup was changed in JBoss Enterprise Application Platform (EAP) 6, the `org.jboss.seam.transaction.transaction` Seam 2.2.x component failed to look up transaction contexts. The respective transaction lookup was added to Seam in JBoss Enterprise Application Platform 5 so that transaction lookup from user applications and the Seam examples now work as expected on *both* JBoss Enterprise Application Platform 5 and 6.

<https://issues.jboss.org/browse/JBPAPP-6924>

Multiple Seam classes have been changed to make `JpaIdentityStore` clusterable. The changes include serialization of the involved classes.

<https://issues.jboss.org/browse/JBPAPP-8159>

Seam component precedence defined in the `components.xml` file was not taken into account when initializing component properties. If a property of the same component was defined more than once in the file, the last occurrence in the file was used, not the one defined in the `<component>` element with the highest value of the `precedence` attribute. This behaviour is now fixed and the property value defined in the `<component>` element with the highest precedence is used during component initialization.

<https://issues.jboss.org/browse/JBPAPP-7957>

When the `<p:piechart>` element was used inside the `<ui:repeat>` element, data in the piechart dataset generated in every `<ui:repeat>` iteration were accumulated with the ones from the previous iterations. The behaviour has been fixed to reset and create the dataset in every iteration, which ensures that correct datasets are now generated.

<https://issues.jboss.org/browse/JBPAPP-7201>

Business interfaces in a Seam Component were stored in the `java.util.HashSet` data structure. This data structure does not guarantee the order of elements, thus the order of business interfaces was nondeterministic. This was fixed by using the `java.util.LinkedHashSet` data structure instead.

<https://issues.jboss.org/browse/JBPAPP-7183>

When using the `<s:token />` tag in JSF, the `javax.faces.ClientToken` cookie that is created sometimes had an empty ("" ) path. The cookie was then not accepted in certain versions of **Windows Internet Explorer** and the following error message was returned:

```
viewId: /myPage.xhtml - No client identifier provided.
```

This update ensures that when an empty path is detected, the cookie path is set to `"/`. As a result, the cookie is accepted by all browsers and the above mentioned error message is no longer returned.

<https://issues.jboss.org/browse/JBPAPP-8376>

The Seambay example failed the functional tests on EAP 6, throwing a `NotLoggedInException`. This was caused by the underlying issue JBPAPP-10075, which was fixed in this release.

<https://issues.jboss.org/browse/JBPAPP-10075>

The Seam web service integration uses the SOAP header for storing the conversation's ID. However the SOAP header is not mandatory and in JBoss Enterprise Platform 6 it is no longer present by default. Therefore any attempts by a Seam web service to process a message without a header failed. To resolve this issue the `SOAPRequestHandler` now adds a header to any message where a header is missing.

<https://issues.jboss.org/browse/JBPAPP-8465>

The Spring example application failed to run on JBoss Enterprise Application Platform 6 because the `cglib.jar` file was signed. This has now been resolved by un-signing the file and the example now runs on JBoss Enterprise Application Platform 5 and 6.

<https://issues.jboss.org/browse/JBPAPP-8915>

The version of Drools distributed as part of JBoss Enterprise Application Platform 5.1.2 was incompatible with Java 7. To resolve this issue the Drools component has been upgraded.

<https://issues.jboss.org/browse/JBPAPP-7520>

The seam-gen tool by default obtained the database driver for the Hypersonic database from `${seam.dir}/lib/hsqldb.jar`. However, that file was moved in an earlier version, resulting in a failed database connection. This issue has been resolved by obtaining the driver from the directory `${jboss.home}/common/lib/hsqldb.jar`.

## Security

<https://issues.jboss.org/browse/JBPAPP-8089>

The `j_password` form field parameter was not filtered out of auditing logs, which caused unintended password exposure in the `audit.log` file. The underlying source code for audit logging has been enhanced to filter out the `j_password` parameter so that it no longer occurs in the `audit.log` file.

<https://issues.jboss.org/browse/JBPAPP-7794>

Login modules have been enhanced so that if invalid configuration options are used, these are logged as warnings. This reduces the time required to diagnose the problem.

## System

<https://issues.jboss.org/browse/JBPAPP-9973>

When two subsequent calls to suspend the hot deployment scanner were made without resuming the scanner in between the calls, any following calls of the `resume()` method were not functional. As a result, it became impossible to resume the scanner. This behavior has been fixed and it is now possible to resume the hot deployment scanner after a series of subsequent suspend calls.

## Transaction Manager

<https://issues.jboss.org/browse/JBPAPP-8073>

If an unexpected failure occurred during execution of the `Synchronization.afterCompletion` method, the original exception thrown after the failure was not logged and the following warning was logged instead:

```
2012-02-08 12:47:31,402 WARN
[com.arjuna.ats.arjuna.logging.arjLoggerI18N]
(WorkerThread#0[127.0.0.1:46304])
[com.arjuna.ats.arjuna.coordinator.TwoPhaseCoordinator_4]
TwoPhaseCoordinator.afterCompletion - returned failure for
com.arjuna.ats.internal.jta.resources.arjunacore.SynchronizationImple@3c
255a5a
```

This behavior prevented exact identification of the failure. With this update, logging has been extended to include such exceptions. The exceptions are now logged, enabling easier identification of failures that occur.

## Web

<https://issues.jboss.org/browse/JBPAPP-10001>

JBoss Web would only start processing requests once it had received notification from the bootstrap process that everything had been successfully booted. If the bootstrap failed, the result was that the application server was running and consuming resources, but not accepting connections. This behaviour has been changed so that JBoss Enterprise Application Platform exits if the bootstrap fails. On exiting it outputs the message **Failed to boot JBoss** and a stack trace. This change brings JBoss Enterprise Application Platform 5 in line with the behaviour of JBoss Enterprise Application Platform 6.

<https://issues.jboss.org/browse/JBPAPP-9561>

Error during startup in `ClusteredSingleSignOn` valve was incorrectly logged as the *Caught exception stopping*. With this update, the problem has been fixed and the error is now logged as *Caught exception starting*.

<https://issues.jboss.org/browse/JBPAPP-8112>

Due to a regression, the compilation of some JSPs with JSF tags (for example `<h:outputText value="\\">`) failed with an exception similar to the following:

```
Caused by: org.apache.jasper.JasperException: /pages/preferences.jsp
According to TLD or attribute directive in tag file, attribute value
does not accept any expressions
```

This happened because of incorrect expression parsing. Now, the regression has been fixed and the problem no longer occurs.

<https://issues.jboss.org/browse/JBPAPP-7210>

It was found that when an AJP message with a request body was received, an unsolicited AJP message containing the first part or the entire request body was sent to the web server under certain circumstances. This injected message could be processed as a new request which would permit an attacker to gain full control over the AJP message and bypass authentication, and lead to information disclosure. With this update, such message injections no longer take place.

<https://issues.jboss.org/browse/JBPAPP-7401>

There was a concurrency issue in Tomcat. When the `javax.servlet.ServletOutputStream.write` method got accessed by multiple threads, it sometimes got called while the previous thread had not yet finished writing of the byte buffer, or when its own buffer had not yet been flushed. This resulted in corrupted output and the following exception was thrown:

```
SEVERE: Servlet.service() for servlet default threw exception
java.io.IOException: Bad file number
    at java.io.FileInputStream.readBytes(Native Method)
    at java.io.FileInputStream.read(FileInputStream.java:199)
    at java.io.BufferedInputStream.read1(BufferedInputStream.java:256)
    at java.io.BufferedInputStream.read(BufferedInputStream.java:317)
    at java.io.FilterInputStream.read(FilterInputStream.java:90)
    at
org.apache.catalina.servlets.DefaultServlet.copyRange(DefaultServlet.java:1968)
    at
org.apache.catalina.servlets.DefaultServlet.copy(DefaultServlet.java:1714)
    at
org.apache.catalina.servlets.DefaultServlet.serveResource(DefaultServlet.java:809)
    at
org.apache.catalina.servlets.DefaultServlet.doGet(DefaultServlet.java:325)
    at javax.servlet.http.HttpServlet.service(HttpServlet.java:690)
    at javax.servlet.http.HttpServlet.service(HttpServlet.java:803)
```

This update modifies the underlying source code and ensures that such concurrency is avoided. The `javax.servlet.ServletOutputStream.write` method can now be accessed by multiple threads without causing the previously mentioned issue.

<https://issues.jboss.org/browse/JBPAPP-7098>

A number of security vulnerabilities were identified in JBoss Web. These have now been resolved in this release and in patches to the previous release. Refer to **CVE-2011-2204**, **CVE-2011-2729**, **CVE-2011-1184**, **CVE-2011-2526** and **CVE-2011-4858** for details of the vulnerabilities.

<https://issues.jboss.org/browse/JBPAPP-6382>

When a `null` reference and a `Number` type was passed to the `coerceToType` method of the `org.apache.el.lang.ELSupport` class, the following error message was thrown:

```
java.lang.IllegalArgumentException: Cannot convert 0 of type class
java.lang.Long to class java.lang.Number
```

The cause of this issue was that the `coerceToNumber` method did not check if the value was of the `Number` type. This test has now been added, resolving this bug.

## Web Services

### <https://issues.jboss.org/browse/JBPAPP-8011>

Previously, when `jbossws.undefi ned.host` was used as the value of the `webServiceHost` property in the `jboss-beans.xml` file, the `REPLACE_WITH_ACTUAL_URL` string was not replaced by the actual address when used in a WSDL definition as the value of the `location` attribute of the `soap12.address` element. This update fixes this issue and the `REPLACE_WITH_ACTUAL_URL` string is now replaced with the appropriate SOAP 1.2 address.

### <https://issues.jboss.org/browse/JBPAPP-7108>

Changes made to resolve a security issue ([JBPAPP-6879](https://issues.jboss.org/browse/JBPAPP-6879)) created a performance regression. The cause of the issue was that in the related change, the `DOCTYPE` declaration in `DOMUtils` was disabled. This has been corrected and the performance regression no longer occurs.

### <https://issues.jboss.org/browse/JBPAPP-6860>

When a JBoss Web Services (WS) client sent a response containing an invalid character, it returned `null` rather than raising an exception. The expected behaviour was that the client stub throw the following exception: `javax.xml.ws.WebServiceException`. The only error reported, logged to `stderr`, was as follows:

```
ERROR [STDERR] [Fatal Error] :2:1462: An invalid XML character (Unicode: 0xffff) was found in the element content of the document.
```

A system property `org.jboss.ws.IgnoreParseError` can now be set on the client side using `JBossWS` to allow exceptions to be propagated to the client instead of returning `null`. If this property is not set, `null` is returned as previously.

### <https://issues.jboss.org/browse/JBPAPP-7498>

When using the SOAP Envelope and accessing the default namespace, a `NullPointerException` would be raised. The root cause of this issue has been resolved, so that this issue will no longer occur.

### <https://issues.jboss.org/browse/JBPAPP-6249>

The `JBossWS-CXF JAXWS` client relies on the `CXF ProviderImpl` which requires a `Bus` instance. Unless a `Bus` was associated with the current thread, the `Bus` retrieval process returned the `Bus` instance that was created for the server. Now a customized `Provider` implementation creates a new `Bus` instance for the client and the `Bus` request returns this `Bus` instance.

### <https://issues.jboss.org/browse/JBPAPP-7900>

An upstream patch applied to the Java Architecture for XML Binding (JAXB) component was found not to be thread safe. The issue was patched and the patched version of JAXB has been included in this release of JBoss Enterprise Application Platform 5.

### <https://issues.jboss.org/browse/JBPAPP-7982>

`JBossWS` previously did a character-for-character comparison of the issuer field of certificates when comparing the client's certificate against the truststore's certificate. It now follows the X509 specification more closely, allowing issuer fields to be considered a match when not an exact

character-for-character match.

<https://issues.jboss.org/browse/JBPAPP-8810>

JBossWS-CFX failed to create a dispatch object when using `javax.xml.ws.Service.createDispatch()` methods involving EPRs (Endpoint Reference) with the following exception:

```
javax.xml.ws.WebServiceException: Cannot find port: null
```

Consequently, the web service invocation failed. This happened due to missing and incorrect code for handling EPR deserialization. With this update, the deserialization code has been provided and corrected, and the problem no longer occurs.

<https://issues.jboss.org/browse/JBPAPP-7112>

When requesting a WSDL or external XML schema from a JBossWS endpoint (e.g. with `?wsdl`), the XML declaration was omitted. This issue has been resolved and the declaration is now included with UTF-8 encoding.

<https://issues.jboss.org/browse/JBPAPP-7461>

The `2.5.4.SP1` version of JBoss Remoting was specified as a dependency in the `pom.xml` file for JBossWS Native stack. This version was outdated compared to the Enterprise Application Platform component matrix, which contained version `2.5.4.SP3`. With this update, the `pom.xml` file has been updated and now contains the correct dependency on version `2.5.4.SP3`.

<https://issues.jboss.org/browse/JBPAPP-9457>

Due to an issue in a DOM utility class, the procedure for finding children of a soap element by their tag name was failing and returned a `NullPointerException` when the soap element had the null namespace. With the update, the parsing process has been fixed and the search process completes successfully.

<https://issues.jboss.org/browse/JBPAPP-9517>

The invocation message could not previously be validated if a WSDL contained multiple nested schemas. This issue has now been resolved and nested schemas are now validated correctly.

<https://issues.jboss.org/browse/JBPAPP-9520>

Previously, a random number was added to each temporary file's name, causing inconsistent naming across instance restarts or reboots, and also in load-balanced environments. This resulted in inconsistencies in load-balanced environments and `OutOfMemory` errors in cases where millions of temporary files were created. With this update, MD5 hashes are used instead of the random numbers, ensuring unique file identification and avoiding the previous issues.

<https://issues.jboss.org/browse/JBPAPP-9545>

The exception when a class name from `@WebService.endpointInterface` is incorrect was difficult to understand. The exception now explains why the error was thrown, making it easier to pinpoint the root cause.

<https://issues.jboss.org/browse/JBPAPP-8118>

A keystore can contain a private key that has a different password to the password of the keystore. To use the key in a web service client, the key-password pair must be defined in the `<key-password>` element in the `jboss-wsse-client.xml` configuration file. Previously, the WS client

or WS endpoint that used the configuration file did not process the `<key-password>` element as the jbossws service failed to pick up the element. The service now detects and picks up the `<key-password>` element as expected and the problem no longer occurs.

<https://issues.jboss.org/browse/JBPAPP-8018>

Web Services Description Language (WSDL) files in the directory `<SERVER>/data/wsd1` were re-generated after each EAP server restart, but existing temporary files were not deleted, resulting in an ever-increasing storage usage. Handling of these files has now been improved, with consistent file naming across restarts avoiding the need to recreate them. The new method uses an MD5 hash to uniquely identify each file.

## Other

<https://issues.jboss.org/browse/JBPAPP-9412>

It was found that when a `.sar` archive that declared a classpath element in its `*-service.xml` file was deployed, the `MANIFEST.MF Class-Path` of the `.jar` files specified in the `.sar`'s classpath were not added to the `.sar`'s classpath. This release has patched the `SARDeployer` such that the classpath is transitive, so if the `sar`'s classpath includes `x.jar` and `x.jar`'s classpath includes `z.jar`, then the `.sar`'s classpath also includes `z.jar` as it should.

<https://issues.jboss.org/browse/JBPAPP-6920>

A syntax error was discovered in the `run.bat` batch file distributed with JBoss Enterprise Application Platform 5.1.1. When run on Windows Server, if `bin` directory was not the current directory, `run.bat` failed to call the associated `run.conf.bat`. The batch file has now been corrected and `run.conf.bat` is called as intended.

<https://issues.jboss.org/browse/JBPAPP-10165>

An infinite number of reconnection attempts can be specified by setting the value of the `reconnectAttempts` property of a message driven bean to `-1`:

```
@ActivationConfigProperty(propertyName="reconnectAttempts",
    propertyValue="-1")
```

However, this was previously functional only with HornetQ. With JBoss Messaging, no reconnection attempts were performed when the `-1` value was used. With this update, the underlying JBoss Messaging source code has been modified to support this value. As a result, an infinite number of reconnection attempts is now performed when the value is set to `-1`.

<https://issues.jboss.org/browse/JBPAPP-7800>

The Apache xml-commons resolver did not correctly convert paths to URLs. Instead, it prefixed the path with `file:` and passed it to the URL constructor. This method was incompatible with Microsoft Windows Server because `file:C:\...` is not a valid file URL. This function has been corrected and the method of converting paths to URLs is now compatible with all supported platforms.

<https://issues.jboss.org/browse/JBPAPP-9156>

An issue was found in an underlying library, in which a `StackOverflow` error could occur while validating a long XML file against a schema. The root cause of this issue has been resolved and the error no longer occurs.



<https://issues.jboss.org/browse/JBPAPP-6327>

Messages for any given queue could not be removed when the messages were paged. The HornetQ core API removed only the in-memory messages but not the paged ones when `jmsQueueControl.removeMessages(null)` was used. This problem has now been solved and both paged and non-paged messages are successfully removed from the message queue.

<https://issues.jboss.org/browse/JBPAPP-9685>

Previous versions of JBoss Enterprise Application Platform 5 were released with a bug in the `log4j.xml` file. This file contained two `jgroups` categories. This prevented correct logging and resulted in empty `cluster.log` files. The extra `jgroups` category has been removed in this release and logging now functions correctly.

<https://issues.jboss.org/browse/JBPAPP-9316>

Oracle's JDK7 changed the way in which the `PropertyEditors` registry was stored. What was previously a global registry became a per `ThreadGroup` registry. EAP 5.2.0 has removed the use of `PropertyEditorManager` and implements a new facility provided by `jboss-common-beans` to accommodate this. Clients whose applications use custom `PropertyEditors` should be aware of this change.

<https://issues.jboss.org/browse/JBPAPP-9642>

The root directory of JBoss Enterprise Application Platform now contains the `version.txt` file with the full name and exact version of the installed product.

<https://issues.jboss.org/browse/JBPAPP-8065>

When a third party JAR or application tried to open a connection and provide a URL that was not a valid URI, the following exception was thrown and the connection was not established:

```
java.lang.IllegalArgumentException: URI is not hierarchical
at java.io.File.<init>(File.java:363)
at org.jboss.net.protocol.file.FileURLConnection.<init>
(FileURLConnection.java:62)
at org.jboss.net.protocol.file.Handler.openConnection(Handler.java:40)
at java.net.URL.openConnection(URL.java:945)
at java.net.URL.openStream(URL.java:1010)
```

This update ensures that when the `org.jboss.net.protocol.file.useURI` system property is set to `false`, the provided URLs is not required to be a valid URI. As a result, the exception is no longer thrown when the property is set to `false`, and the connection can be successfully established.

<https://issues.jboss.org/browse/JBPAPP-7609>

Classloading classes did not cache the results of `getResourcesLocally()` as they do for `getResourceLocally()`, resulting in `ClassLoader.getResources()` running slowly. This could cause performance issues if it was called often, as Java's `ServerFinder` would do if its results were not cached. To resolve this issue, caching has now been implemented in classloading classes, resulting in better overall performance.

## APPENDIX A. REVISION HISTORY

<b>Revision 5.2.0-101.400</b> Rebuild with publican 4.0.0	<b>2013-10-31</b>	<b>Rüdiger Landmann</b>
<b>Revision 5.2.0-101</b> Final release notes for JBoss Enterprise Application Platform 5.2.0 GA.	<b>Wed 23 Jan 2013</b>	<b>Russell Dickenson</b>

# INDEX