Oracle® Fusion Middleware

Release Notes for Oracle WebLogic Server

14c (14.1.2.0.0)
F43296-01
December 2024

Oracle Fusion Middleware Release Notes for Oracle WebLogic Server, 14c (14.1.2.0.0)

F43296-01

Contents

Preface

Oracle WebLogic Server 14c (14.1.2.0.0) Release Notes summarize release information related to new features or enhancements, resolved issues, general issues and workarounds, deprecated and removed functionality, and more.

Audience

This document is intended for all users of Oracle WebLogic Server 14c (14.1.2.0.0).

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.

Convention	Meaning
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Introduction

This chapter introduces the release notes for Oracle WebLogic Server 14c (14.1.2.0.0). This chapter includes the following topics:

Latest Release Information

This document is accurate at the time of publication. Oracle will update the release notes periodically after the software release. You can access the latest information and additions to these release notes on the Oracle Help Center at: `https://docs.oracle.com/en/middleware/fusion-middleware/weblogic-server/14.1.2/index.html`.

Purpose of This Document

This document contains the release information for Oracle WebLogic Server 14c (14.1.2.0.0). It describes the differences between the software and its documented functionality.

Oracle recommends that you review its contents before installing or working with the product.

System Requirements and Specifications

Oracle WebLogic Server installation and configuration do not complete successfully unless the hardware and software pre-requisite requirements are met before installation.

For the latest information on supported system requirements and system configuration, see:

- System Requirements and Specifications
- Oracle Fusion Middleware Supported System Configurations

Product Documentation

For complete documentation, go to Oracle WebLogic Server.

Oracle Support

Oracle customers who have purchased support have access to electronic support through My Oracle Support at https://support.oracle.com.

Licensing Information

For the latest information on Oracle Fusion Middleware Licensing, see Licensing Information User Manual.

New Features in This Release

For information about the new features in this release, see What's New in This Release in *What's New in Oracle WebLogic Server*.

Deprecated Features

For information about the deprecated features, see Deprecated Features in *What's New in Oracle WebLogic Server*.

2

Known Issues and Workarounds

This chapter describes issues associated with Oracle WebLogic Server 14.1.2.0.0.
This chapter includes the following topics:

General Issues and Workarounds

This section describes the following issues and workarounds:

Multi-Byte Characters Display Incorrectly in Filenames When Using Safari

Issue

Impacted Platforms: All

When using the Safari browser to download content, if a filename contains multi-byte characters, the characters are displayed as '------' in the filename.

Workaround

Set `UseHeaderEncoding` to `true` on the Managed Server. Use the following WLST commands to do so:

```
connect("admin_name", "admin_password", "t3://localhost:port")
edit()
startEdit()
cd("Servers/server_name/WebServer/server_name")
set("UseHeaderEncoding", "true")
save()
activate()
exit()
```

In Turkish Locale, MDS Initialization Fails

Issue

Impacted Platforms: All

Any applications that use an MDS repository cannot be deployed or run with the JAXB version bundled with Oracle WebLogic Server as null values are returned for attributes named `id`

Workaround

Start the server in English locale.

Administration Server Reports a 'Too Many Open Files' Message on the EM Console

Issue

Impacted Platforms: Linux

The WebLogic Server Administration Server reports a `Too Many Open Files` message on the Enterprise Manager (EM) console when the maximum number of file descriptors configured for the Administration Server is less than 65535.

Workaround

Execute the following command to determine the maximum number of file descriptors currently configured:

```
cat /proc/sys/fs/file-max
```

If the value is less than 65535, perform the following steps:

1. Edit the file /etc/security/limits.conf with root permission:

   ```
   > sudo vi /etc/security/limits.conf
   ```

2. Append the following two lines, using a value of 65535 or greater:

   ```
   *               soft    nofile          65535
   *               hard    nofile          65535
   ```

3. Start a new terminal session.

4. Execute the `limit descriptors` command to verify that descriptors has been increased to the specified value (at least 65535).

   ```
   > limit descriptors
   descriptors  65535
   ```

Installation Requirements if Using Coherence with Maven

Issue

Impacted Platforms: N/A

Coherence users who do not have a dependency on Oracle WebLogic Server and want to use Maven should use the standalone Coherence installer.

Coherence users who do have a dependency on Oracle WebLogic Server and want to use Maven must choose the **WebLogic Server** or **Complete with Examples** installation options. Do not choose the **Coherence Installation** option.

Workaround

N/A

Default WebLogic Server Message Prefix Will Change

Issue

Impacted Platforms: All

The default WebLogic Server message prefix will change from BEA to WL in a future release of WebLogic Server.

Workaround

N/A

Clustering Issues and Workarounds

This section describes the following issues and workarounds:

Threads Are Blocked on Cluster Messaging in Unicast Mode

Issue

Impacted Platforms: Linux

When using Unicast mode for cluster communication, many threads are blocked on cluster messaging, which may result in cluster members having difficulty sending heartbeat messages. In this situation, some cluster members drop out from the cluster and may take some time to rejoin the cluster.

Workaround

N/A

Impact of Minimum and Maximum Dynamic Cluster Size Constraints

Issue

Impacted Platforms: All

To support elasticity, Oracle WebLogic Server 12.2.1 introduced the following configurable constraints on the minimum and maximum size of a dynamic cluster:

- `MinDynamicClusterSize` (default=1)
- `MaxDynamicClusterSize` (default=8)

Additionally, the `MaximumDynamicServerCount` attribute is deprecated and replaced with the `DynamicClusterSize` attribute. The value of this attribute is validated against the previously mentioned minimum and maximum constraints. As a result, some existing user configurations and/or scripts may fail. If this occurs, you need to update the `MaxDynamicClusterSize` setting appropriately so that the `DynamicClusterSize` value is within the limits.

Workaround

N/A

HTTP 503 Error for HTTP POST Requests During Cluster Scale Down or Failover

Issue

Impacted Platforms: All

When `HTTP POST` requests are serviced by clustered Oracle WebLogic Server applications, which are configured with a load balancer such as Oracle HTTP Server, a web server using an Oracle WebLogic Server Proxy Plug-In, or Oracle Traffic Director, an HTTP 503 error can occur. If a POST request has been sent to a Oracle WebLogic Server clustered Managed Server that is shutting down, and if the server is unable to complete the request, or if the result of the request is unknown, then the load balancer is required to return an HTTP 503 error. When Oracle Traffic Director is used with dynamic clusters, and the cluster is being scaled down, Oracle WebLogic Server notifies Oracle Traffic Director of the impending graceful shutdown of the servers and attempts to route requests to the remaining servers in the cluster. However, there may be a brief downtime in between the graceful shutdown operation and before Oracle Traffic Director redirects the HTTP traffic, when some HTTP requests may receive the 503 error.

Workaround

N/A

Configuration Issues and Workarounds

This section describes the following issues and workarounds:

Initialize SLF4J Framework

Issue

After upgrading Oracle WebLogic Server to 14c (14.1.2.0), the `slf4j` framework is not initialized.

Workaround

In Oracle WebLogic Server 14c (14.1.2.0), the `slf4j` framework is located in `$ORACLE_HOME\oracle_common\modules\thirdparty\apache-maven_bundle\3.9.4.0.0\apache-maven-3.9.4\lib`. Initialize the `slf4j` framework from this location.

'No SLF4J providers were found' Warnings

Issue

After upgrading Oracle WebLogic Server to 14c (14.1.2.0.0), `slf4j-api` is added to the `weblogic.jar` file to satisfy third-party dependencies, however, adding logging implementations to the classpath now is being left up to you. If you have `slf4j-api` on the classpath *without* a logging implementation, then you will get warnings on standard output, like the following:

```
SLF4J(W): No SLF4J providers were found.
SLF4J(W): Defaulting to no-operation (NOP) logger implementation
SLF4J(W): See https://www.slf4j.org/codes.html#noProviders for further details.
```

Workaround

You should add *$MW_HOME*`/wlserver/modules/org.slf4j.slf4j-nop.jar` to your classpath or add one of the logging implementation JAR files. For the SLF4J provided explanation, see https://www.slf4j.org/codes.html#noProviders.

ASProvWorkflowException Occurs When Creating a WebLogic Domain

Issue

Impacted Platforms: All

In rare cases, if your installation environment contains existing JAVA_OPTIONS prior to starting a Fusion Middleware product installation, these may cause an `ASProvWorkflowException`, preventing the domain from being created.

Workaround

Prior to starting the Fusion Middleware product installation, clear the existing JAVA_OPTIONS. If you have an application in the environment that uses these JAVA_OPTIONS, the application may not work after clearing the options. In this case, save the existing JAVA_OPTIONS to a text file and investigate alternatives for running your application.

Use the -Dfile.encoding Property When Running WLST in a Non-English Locale

Issue

Impacted Platforms: MS Windows

WLST can be run with localized messages by setting the desired locale. You should be aware of the following issue when running WLST in a non-English locale.

On Windows operating systems, if a DOS command window's active code page is different from the system's local (ANSI) code page, you must add the `-Dfile.encoding=<DOS window's active code page>` property to the WLST process when starting WLST by using a DOS command window. This property changes the default character set for the Java process. For example:

- The active code page for a DOS window is 850. This value can be achieved by issuing the `chcp` command in the WLST command window.
- The system's local (ANSI) code page is 1250. You can determine the system's local code page by viewing the value of the `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\NLS\CodePage\ACP` key in the Windows registry. Files that are created by standard Windows editing tools (such as Notepad or Wordpad) are encoded in this way.

Workaround

In this situation, you can start WLST as follows:

```
set WLST_PROPERTIES="-Dfile.encoding=cp850"

$WL_HOME%\common\bin\wlst.cmd
```

Configuration Tools Can Fail if the Oracle WebLogic Server Installation Path Contains Spaces

Issue

Impacted Platforms: MS Windows

On some Microsoft Windows platforms, the Oracle WebLogic Server configuration tool commands (including `wlst`, `config`, `pack`, and `unpack`) can fail if the Oracle WebLogic Server installation path contains a space. In this case, the command may fail with a `java.lang.ClassNotFoundException` error, where the class is derived from the portion of the installation path after the space. The commands fail if short file name generation is disabled in the Windows registry.

Workaround

You must enable short name generation in the Windows registry to ensure that spaces are handled properly by the configuration tools. To enable short name generation:

1. Run `regedit`.
2. Navigate to the `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\FileSystem` folder.
3. Double-click `NtfsDisable8dot3NameCreation` and set its value to `0`.
4. Reboot for the change to take effect.

Directory For a Non-Existent Server Name Is Created

Issue

Impacted Platforms: All

If you attempt to connect to the WebLogic Server Administration Server with a non-existent server name, a directory for the non-existent server name is created under the *domain_name*/servers directory.

Workaround

Specify a valid server name when starting the Managed Server.

Abnormal Behavior in Terminal Window After Entering WebLogic Password

Issue

Impacted Platforms: Linux

After pressing **Ctrl-C** to terminate the `startManagedWebLogic.sh` process immediately after entering the WebLogic password, abnormal behavior may be experienced in the terminal window. For example, when pressing Return, the prompt is tabbed instead of going to the next line, and any characters that are entered at the prompt are not displayed in the terminal.

Workaround

Either close the current xterm and start a new one, or enter `stty echo` into the xterm.

Creating and Updating Domains Takes Too Long

Issue

Impacted Platforms: Linux

It can take a long time to create or update Oracle WebLogic Server domains when:

- Using any of the prescribed domain creation or update methods, such as the Configuration Wizard or WLST. The method may appear to hang or take significantly longer than normal when used on new Linux hosts. This issue occurs due to the lack of entropy on the host to initialize Java security.
- Using the Oracle WebLogic Server Configuration Wizard to create or update a domain.
- Using WLST to create or update a domain.

Workaround

Set the *CONFIG_JVM_ARGS* environment variable to the following value:

```
-Djava.security.egd=file:/dev/./urandom
```

Password Field is Not Editable When Configuring a New Domain

Issue

Impacted Platforms: Linux

On Linux systems, when creating a new domain using the Oracle Fusion Middleware Configuration Wizard, the **Password** and **Confirm Password** fields are sometimes not editable, and you cannot enter a password to create a domain.

Workaround

There are two ways to work around this issue:

- To work around the issue each time it happens, click the Close Window **X** button in the upper right corner of the Configuration Wizard. In the confirmation dialog box, click **No** to return to the Configuration Wizard. You can then enter and confirm the password for the domain.
- To fix this issue permanently:
  1. Kill all scim processes. For example:

     ```
     kill `pgrep scim`
     ```

  2. Modify (or create) the `~/.scim/config` file to include the following line (case-sensitive):

     ```
     /FrontEnd/X11/Dynamic = true
     ```

  3. If you are running VNC, restart the VNC server.
  4. Run the Configuration Wizard again.

Administration Server Memory Consumption and JMX Notifications

Issue

Impacted Platforms: All

The Domain Runtime MBean Server is a federated MBean server with connections to all Managed Server Runtime MBean Servers in the domain. The federation architecture performs well with queries. However, when JMX notifications are added to MBeans, the Domain Runtime MBean Server can consume large amounts of memory.

When JMX notifications are used, two cases exist that cause the Administration Server to keep copies of all JMX object names registered in all Runtime MBean Servers running in all Managed Servers in the domain:

- At the Oracle WebLogic Server level, to simulate the unregistered MBean notifications when a Managed Server shuts down.
- At the JDK JMX client notification layer.

The likelihood of encountering this issue increases when both of the following conditions exist:

- EM Fusion Middleware Control is being used to manage large domains, as it adds notification listeners to the Domain Runtime MBean Server.
- Fusion Middleware products that significantly increase the number of JMX runtime MBeans are included in the domain. This would include any product with MBeans that are registered in WebLogic Server Runtime MBean Server instances running in the domain; that is, in the Administration Server as well as all Managed Servers. (These products include Coherence, SOA Suite, OSB, and so on.)

Workaround

Disable the `managed-server-notifications-enabled` attribute. This configuration attribute disables the ability to define notifications on MBeans that are contained in the Managed Servers Runtime MBean Servers (these MBeans contain a `Location=key` in the ObjectName).

If Managed Server notifications are disabled, then the two sets of ObjectNames for MBeans contained in the Oracle WebLogic Server and JDK components will not be kept. Notifications

listeners can still be defined on the MBeanServerDelegate and on MBeans contained in the local Domain Runtime MBean Server. However, notifications listeners cannot be added to the non-local MBeans.

Coherence Cache Override Not Working

Issue

Impacted Platforms: All

If the Oracle WebLogic Server Configuration Wizard (`config.sh`) is used to create a domain and the WebLogic Coherence Cluster Extension template is specified, then a Coherence cluster will be defined. The Coherence cluster will be associated with any Managed Server or Oracle WebLogic Server cluster that is also created by the Configuration Wizard. If no Managed Server or Oracle WebLogic Server cluster is created, then the Coherence cluster will be associated with the Administration Server. This association between the Coherence cluster and the servers is not completely defined using the Oracle WebLogic Server configuration tool, which results in the Coherence cache configuration override file not being detected by the Coherence cluster. Note that this issue only occurs if you are using the cache override feature.

This will perform a complete association between the Coherence cluster and the targeted servers, which is required to detect and utilize the specified Coherence cluster cache configuration override file.

Creating a Managed Server Domain from a Template Causes Error

Issue

Impacted Platforms: All

Specify `-managed=true` when creating a Managed Server domain directory from a template. If you do not specify `-managed=true`, the Managed Server will fail to boot because it does not have the correct set of files in the security directory.

Workaround

N/A

Changing Domain From Development To Production Mode Does Not Change Start Scripts

Issue

Impacted Platforms: All

When you change a domain from development mode to production mode:

- The domain start scripts (and the value of the `-Xverify` flag) do not change.
- The `boot.properties` file continues to be in use.

Workaround

In production mode domains:

1. The value of the `-Xverify` flag in the start scripts needs to be changed from `none` to `all`.

2. The `boot.propeties` file needs to be removed. For more information, see Development and Production Mode in *Understanding Domain Configuration for Oracle WebLogic Server*.

Error Occurs When Running WLST Script or Command

Issue

Impacted Platforms: All

When running a WLST script or command, a `TypeError: state(): 1st arg can't be coerced to String` error occurs. This error occurs because a WLST class name `server` is used as a variable name in a WLST command. For example, in the following command, the value for the variable `server` will be replaced with its class name during classloading and will therefore cause this error:

```
state(server,'Server')
```

Workaround

Use one of the following workarounds:

- Include the `-Dpython.cachedir.skip=true` parameter when starting WLST.
- Change the reserved string name to another string. For example, you can change the string name `server` to `srvr` to resolve the issue.

BI Cluster1 is Down on a Pure IPv6 Deployment After Configuration

Issue

Impacted Platforms: N/A

This is on a pure IPv6 environment. After installing WLS and configuring BI, the bi_cluster is down. This was not reported by the Configuration Wizard.

Avoid Using the Default Configuration for Oracle WebLogic Server When Running with the ODA Cluster Configuration

Issue

Impacted Platforms: Linux

Using the current default multicast addresses blocks Oracle Clusterware if Oracle cluster is sharing the same subnet as Oracle WebLogic Server.

Workaround

Oracle recommends that you avoid using the current default configuration setting for Oracle WebLogic Server because this configuration can block Oracle Clusterware located in the same subnet.

Core Server and Core Work Manager Issues and Workarounds

This section describes the following issues and workarounds:

Using IPv6-Formatted Addresses

Issue

Impacted Platforms: All

When using an IPv6-formatted address for Oracle WebLogic Server, the URL should include square brackets ('[' and ']') for the host address. Otherwise, WLST may fail to connect to the running server.

Workaround

Add square brackets to the host address. For example:

```
t3://[fe80:0:0:0:203:baff:fe2f:59e5]:9991
```

Object State is Not Retained After Renaming a Field

Issue

Impacted Platforms: All

When FastSwap is enabled in a J2EE application, you can make certain types of changes to Java classes during development and expect to see the change without re-deploying, with all instance states of the Java object being retained.

One type of change that does NOT retain the object state is when you change a field name. The field name change is treated as follows:

- The field with the old name is deleted.
- The field with the new name is added.

In this case, any state in the old field is not carried over to the renamed field.

Using the Workshop or FastSwap Ant task, you may see a `FastSwap operation completed successfully` message, even when an instance field name change causes a value to reset.

Workaround

You should expect an instance value to be reset when you change a field name.

Servers Configured to Listen on a Host Name Are Listening on a Different Host Name After Startup

Issue

Impacted Platforms: All

When using a host name to specify configuring the listen address on the WebLogic Server Administration Server or a Managed Server, machines that are configured with multiple Ethernet cards may listen on a different host name after startup. For example:

- The machine has 3 Ethernet cards
- Card 1 is mapped to `hostname1-s` (DNS registered host name)
- Card 2 is mapped to `hostname1-i` (DNS registered host name)
- Card 3 is mapped to `hostname1` (actual node's host name)
- You configure the server to listen on `hostname1`
- After starting the server, it is listening on `hostname1-s` because Windows resolves the actual node's host name to the first enabled Ethernet card address

Workaround

Use one of the following three workarounds for this issue:

1. Use the IP address, instead of the host name, as the listen address of the WebLogic Server Administration Server. On Managed Servers, use the IP address as the listen address, or configure the actual physical host name to the first Ethernet card in the machine.
2. Add the following entry to the C:\Windows\system32\drivers\etc\hosts file on the machine:

   ```
   <ip_address> <hostname>
   ```
3. Change the order of the network cards in the machine so that the card with the actual node's host name is Card 1.

Multicast Traffic Observed to be Unreliable During or After a Network Partition

Issue

Impacted Platforms: Linux

During or after a network partition that causes a server migration to take place, multicast traffic has been observed to be unreliable. For example, one node may be receiving multicast traffic, but traffic originating from this node is not received on other nodes in the network. As a result, the migrated servers are not added to the cluster because their heartbeats were not received.

Workaround

Currently, the only known workaround is to use unicast cluster messaging.

No Java DB Leasing Script or Support

Issue

Impacted Platforms: All

Oracle WebLogic Server does not support Java DB for migration. There is no leasing script available in the *WL_HOME*/server/db directory for Java DB.

Workaround

No workaround for this issue.

JSON and Boolean Collections Support Limited

Issue

Impacted Platforms: All

In FMW 14.1.2.0.0, JSON and Boolean collections are not supported with Oracle Database Version 23ai and SODA JARs version 1.1.7.3.

Workaround

N/A

Cannot Use SQL and SODA API Together for JSON Collection

Issue

Impacted Platforms: All

"TABLE OR VIEW DOES NOT EXIST" is reported when SQL Query (using Statement or PreparedStatement) is used immediately after creating a JSON collection using WLS or SODA API JARs.

Workaround

When creating a JSON collection using SODA APIs, also use SODA APIs for DML or DQL operations.

Alternatively, create a JSON collection without using the SODA API, and then perform DML or DQL operations with SQL or SODA APIs.

Managed Server Fails to Start When Added to Multicast Cluster

Platform: All

If you have a cluster containing at least two Managed Servers, and the messaging mode of the cluster is set to multicast, then the Managed Servers in the cluster will fail to start.

Issue

Impacted Platforms: All

If you have a cluster containing at least two Managed Servers, and the messaging mode of the cluster is set to multicast, then the Managed Servers in the cluster will fail to start.

Workaround

To workaround this issue, add the following property to `JAVA_OPTIONS` in the startup script `startWebLogic.sh`. For example:

```
export JAVA_OPTIONS="${JAVA_OPTIONS} -Djava.net.preferIPv4Stack=true"
```

High Number of Application Threads May Cause a Server to Stall

Issue

Impacted Platforms: All

A server JVM appears to have stalled; and a thread dump of this JVM reveals that almost all `weblogic.kernel.Default` threads appear to be stalled in calls such as `wait-for-data` or `wait-for-prepare-acks`.

Workaround

- Disable the Work Manager enhanced increment advisor by specifying the `-Dweblogic.UseEnhancedIncrementAdvisor=false` system property on the server command line.
- If the previous workaround does not work, perform other workarounds listed below:
  - Set the `FEJmsDispatcher` and `BEJmsDispatcher` to have a minimum constraint of 1 and a maximum constraint of 6, while keeping `Dweblogic.UseEnhancedIncrementAdvisor=false`.
  - Tune applications to use fewer threads each. For examples related to thread management for Message-Driven Beans (MDB), see Tuning Message-Driven Beans in *Tuning Performance of Oracle WebLogic Server*
  - Set the `-Dweblogic.threadpool.MinPoolSize=NNN` property on all server JVMs in a cluster to a value that is 20% higher than the current number of `weblogic.kernel.Default` threads in the thread dump. If you are not sure of the NNN count, try 100, and if that does not work, try 150, and so on. Note that configuring too many minpool threads can cause very poor performance *or even an 'out of memory'*.
  - Set a JTA min-threads-constraint of 20 or 30 on all servers in all involved clusters for the `JTACoordinatorWM` Work Manager. You can do this setting using the `-Dweblogic.transaction.jta.coordinator.wm.min.constraint=YYY` system property.

Enterprise Beans Can be Accessed by Remote Clients When Annotated with @Local

Issue

Impacted Platforms: N/A

As per expected behavior, if the bean class is annotated with `@Local` (or neither with `@Local` nor `@Remote`), and if one of the implemented interfaces is extended with the `java.rmi.Remote` interface, then the interfaces should be accessed only by the local client, not by the remote client.

However, in 12.2.1, the interface can be accessed by the remote client when annotated with `@Local`.

Workaround

If you want access to an EJB to be restricted to local clients only, you should *avoid* both of the following in the EJB implementation:

- Using the `@Local` annotation
- Extending the `java.rmi.Remote` interface

Data Source Issues and Workarounds

This section describes the following issue and workaround:

- Data Source Validation Fails for SQL Server When Specifying a Port Number

Data Source Validation Fails for SQL Server When Specifying a Port Number

Starting from Oracle WebLogic Server 12.2.1, when you connect to a SQL Server named instance with a port number specified in the connection URL, the DataDirect JDBC driver for SQL Server returns the following error:

```
[FMWGEN][SQLServer JDBC Driver]Conflicting connection information. When the
instance name is specified, it is invalid to specify the port number.
```

The data source fails to deploy.

Workaround

Update the JDBC descriptor file to set the `allowPortWithNamedInstance` property to `true` or remove the port number from the URL.

WLST scripts must be updated by the administrator to set the `allowPortWithNamedInstance=true` property to `true`. The following sample URLs are valid:

```
jdbc:weblogic:sqlserver://host\
\INSTANCE:1433;DatabaseName=db;allowPortWithNamedInstance=true"
jdbc:weblogic:sqlserver://host:1433;DatabaseName=db"
jdbc:weblogic:sqlserver://host\\INSTANCE;DatabaseName=db"
```

Dependency Injection Issues and Workarounds

This section describes the following issues and workarounds:

- AroundInvoke Interceptor Methods Apply to MDB Methods Called by Containers
- BeanManager Does Not Contain a ValidatorFactory Bean in Lifecycle Events
- CDI Enabled EAR with Non-CDI Enabled WAR Does Not Work Correctly
- CDI Treatment of RAR Changes in 12.2.1

AroundInvoke Interceptor Methods Apply to MDB Methods Called by Containers

Issue

For Interceptors declared using interceptor bindings, usually the `AroundInvoke` interceptor methods do not apply to MDB methods called by containers, such as `setMessageDrivenContext()`, `ejbRemove()` and so on. From this release of Oracle Weblogic Server 14.1.1.0.0, `AroundInvoke` interceptor methods apply to these methods as well. This behavior change is caused by Weld `3.0.x (x>1)`, which is integrated in Oracle WebLogic Server 14.1.1.0.0 for CDI.

Workaround

Add conditional judgement in `AroundInvoke` to skip these containers called *`methods`*.

BeanManager Does Not Contain a ValidatorFactory Bean in Lifecycle Events

Issue

Impacted Platforms: N/A

If a CDI extension is observing the lifecycle events AfterBeanDiscovery or AfterDeploymentValidation, an attempt to get a `javax.validation.ValidatoryFactory` bean instance will fail. For example:

```
public class MyExtension implements Extension {
     void checkValidatorFactoryBean(@Observes AfterBeanDiscovery abd, BeanManager bm) {
          Set<Bean<?>> validatorFactoryBeans
          = bm.getBeans(ValidatorFactory.class,
          new AnnotationLiteral<Default>() {});
     if (validatorFactoryBeans.isEmpty()) {
          throw new RuntimeException("Container provided BeanManager doesn't contain a
bean of javax.validation.ValidationFactory");
          }
     }
}
```

The example code above will cause a `RuntimeException` because the BeanManager cannot get a ValidatorFactory bean instance. This occurs because the ValidatorFactory and Validator bean types were removed from the built-in beans starting in CDI 1.1. In CDI 1.1 and later, ValidatorFactory and Validator beans are defined by the following extension: `org.hibernate.validator.internal.cdi.ValidationExtension`.

Note that when called during invocation of an AfterBeanDiscovery event observer, this method will only return beans discovered by the container before the AfterBeanDiscovery event is fired.

Workaround

N/A

CDI Enabled EAR with Non-CDI Enabled WAR Does Not Work Correctly

Issue

Impacted Platforms : N/A

If an application contains both a CDI enabled EJB module and a non-CDI enabled web module, events defined with the following qualifier are not sent:

```
@Initialized(ApplicationScoped.class) and @Destroyed(ApplicationScoped.class)
```

Workaround

Place an empty `beans.xml` file in the WEB-INF directory of the WAR file.

CDI Treatment of RAR Changes in 12.2.1

Issue

Impacted Platforms: N/A

For CDI 1.0 in Oracle WebLogic Server, RAR archives are treated as CDI bean archives if the `beans.xml` file is present in the META-INF directory, and the embedded library JAR classes are included as part of that bean archive. As of Oracle WebLogic Server 12.2.1, each archive, including embedded library JARs, is individually considered as a candidate bean archive based on the presence of the `META-inf/beans.xml` file or at least one class with a bean-defining annotation.

Therefore, to reproduce the previous behavior, the embedded library JARs must have either a META-INF/beans.xml entry or at least one class with a bean-defining annotation.

Workaround

N/A

Deployment Issues and Workarounds

This section describes the following issues and workarounds:

security-permission Element is Not Available in weblogic-application.xml

Issue

Impacted Platforms: All

The `security-permission` element is available in the `weblogic.xml` and `weblogic-ejb-jar.xml` deployment descriptors but is not available in the `weblogic-application.xml` descriptor. Therefore, in an Enterprise application, you can apply security policies only to JAR files that are EJBs or web applications.

Workaround

N/A

Extraneous String Values Interpreted as File Specification

Issue

Impacted Platforms: All

The `weblogic.Deployer` tool interprets any extraneous string values between command-line arguments as a file specification. For example, consider the following command:

```
java weblogic.Deployer -activate -nostage true -name myname -source
c:\myapp\mymodule
```

In this case, the tool attempts to activate a file specification named `true` because the `-nostage` option takes no arguments and `true` is an extraneous string value.

Workaround

N/A

restore Method Does Not Update the DConfig Bean with Plan Overrides

Issue

Impacted Platforms: All

The `restore` method does not correctly update the DConfig Bean with the plan overrides. For example, consider the following steps:

```
DeployableObject dObject =
   WebLogicDeployableObject.createDeployableObject(new File(appName));
DeploymentConfiguration dConfig =
   WebLogicDeploymentManager.createConfiguration(dObject);
dConfig.restore(new FileInputStream(new File(plan)));
```

In this case, the plan does not override the DConfig Bean correctly.

Workaround

Specify the plan when initializing the configuration for the application. For example:

```
helper = SessionHelper.getInstance(
    SessionHelper.getDisconnectedDeploymentManager());
helper.setApplication(app);
helper.setPlan(new File(plan));
helper.initializeConfiguration();
```

Deployment Task Fails When a Large Application File Is Deployed

Issue

Impacted Platforms: All

When a large application file is deployed using the `upload` option, the deployment task fails with the following error:

```
java.lang.OutOfMemoryError: Java heap space
```

To resolve this issue, a new system property, `weblogic.deploy.UploadLargeFile`, has been added. If you see this issue, include this flag in the `java` command you use to launch a deployment client.

If you are using the Oracle WebLogic Server patch releases 9.2 MP2, 9.2 MP3,10.0 MP1, 10.0 M2, 10.3, 10.3.1, 10.3.2, or 10.3.3, this flag is not needed.

Workaround

N/A

Attempting to Redeploy an Application Fails if the Application is Already Deployed Using a Different Source File Location

Issue

Impacted Platforms: Linux

If you initially deployed an application using one source file location, then attempt to redeploy the application using a new location for the source file, the deployment fails with the following exception:

```
New source location <new_source_file_path> cannot be configured deployed to
configured application, <application_name>. The application source is at
original_source_file_path. Changing the source location is not allowed for a
previously attempted deployment. Try deploying without specifying the source.
```

This is due to a Oracle WebLogic Server deployment restriction. Once you specify the source file for a deployment, you cannot change it on a redeployment.

Workaround

Undeploy the application before attempting to redeploy it using a new source file location.

Relevant Output Message Not Displayed

Issue

Impacted Platforms: All

If you create an application and deploy it to a target, and then try to deploy that same application to that same target, no relevant output message is displayed to inform you that your application is already deployed to that particular target. This occurs because when the application is deployed the second time it is considered to be the equivalent of a redeploy.

Workaround

N/A

Developer Experience Issues and Workarounds

This section describes the following issue and workaround:

- Users Need to Set BEA_HOME System Property While Using Appc for Pub-Sub Modules

Users Need to Set BEA_HOME System Property While Using Appc for Pub-Sub Modules

Issue

Impacted Platforms: All

An error occurs when using the appc Maven plug-in after installing Oracle WebLogic Server Maven artifacts to the local repository using the Maven synchronization plug-in.

Workaround

Oracle WebLogic Server pub-sub libraries rely on the `BEA_HOME` system property to resolve compiler issues. Set the `BEA_HOME` system property while running appc on pub-sub applications for compilation to resolve these dependencies.

EJB Issues and Workarounds

This section describes the following issues and workarounds:

- Invoking Remote EJB Clients
- Primary Key in Oracle Table is CHAR
- No Annotation that Enables Creation of a Clusterable Timer
- Kodo's MappingTool Cannot Generate Schemas
- Extensions to the JPA Metadata Model can be Specified Only Through Annotations
- Lookup Method Injection Not Supported by Spring
- Deserializing a JDO PersistenceManagerFactory in a Managed Environment May Fail
- Indexes Not Always Created During Schema Creation
- OpenJPA Throws an Exception When @Id Fields are Also Annotated as @Unique
- Cache Hit and Miss Counts May Rise Unexpectedly
- Open JPA Tries to Create a Table Even if the Table Exists
- EJB Applications Fail During Serialization
- EJB Handle Serialization Fails When Using IIOP to Invoke 11g EJB
- WebLogic Client Cannot Use IIOP to Look Up EJB 3.x Specified with the Remote Annotation

Invoking Remote EJB Clients

Issue

Due to per domain CA changes in 14.1.2.0.0, when invoking remote EJB clients, you must set the `-Dweblogic.RootDirectory` system property or import the server certificate into the client trust keystore. In previous releases, this was not required when invoking remote EJB clients because common `DemoTrust.jks` and `DemoIdentity.jks` keystores were used.

Workaround

To specify the system property `-Dweblogic.RootDirectory`, the EJB client should have access to the domain directory. If the EJB client is running on a different host from the WebLogic Server host, then you should import the WebLogic Server certificate into the client trust keystore.

Primary Key in Oracle Table is CHAR

Issue

Impacted Platforms: All

The primary key in an Oracle table is a CHAR but the query field in the SQL table is a VARCHAR2.

Workaround

Change the database schema from CHAR to VARCHAR2. Using CHAR as a primary key is not recommended for the Oracle database.

No Annotation that Enables Creation of a Clusterable Timer

Issue

Impacted Platforms: All

There is no annotation for EJB3 beans or `Ejbgen` that enables creation of a clusterable timer.

Workaround

Create a `weblogic-ejb-jar.xml` file and add the `<timer-implementation>` element and corresponding values in the file.

Kodo's MappingTool Cannot Generate Schemas

Issue

Impacted Platforms: All

Kodo's MappingTool cannot generate schemas for classes that use BLOBs in their primary key. BLOBs can be used in a primary key, but the schema must be defined manually. Note that support for BLOB columns in primary keys is not mandated by either the JDO or JPA specifications.

Workaround

None.

Extensions to the JPA Metadata Model can be Specified Only Through Annotations

Issue

Impacted Platforms: All

Extensions to the JPA metadata model can be specified only through annotations and not by using a structure similar to the `orm.xml` file defined by the specification.

Workaround

To specify Kodo-specific metadata for your object model, do one of the following:

- Use the Kodo-specific annotations.
- Convert the XML-based metadata to the JDO metadata format, which does support XML specification of extensions.

Lookup Method Injection Not Supported by Spring

Issue

Impacted Platforms: All

The WebLogic Spring injection extension model doesn't support lookup method injection.

Workaround

N/A

Deserializing a JDO PersistenceManagerFactory in a Managed Environment May Fail

Issue

Impacted Platforms: All

Deserializing a JDO `PersistenceManagerFactory` in a managed environment may fail. The exception states that the `javax.jdo.PersistenceManagerFactoryClass` property is missing. Note that serializing `PersistenceManagerFactory` should not generally be necessary in a managed environment.

Workaround

None.

Indexes Not Always Created During Schema Creation

Issue

Impacted Platforms: All

Indexes declared at the class level are not always created during schema creation.

Workaround

Create the indexes manually after running the schema generation tools.

OpenJPA Throws an Exception When @Id Fields are Also Annotated as @Unique

Issue

Impacted Platforms: All

OpenJPA throws an exception when `@Id` fields are also annotated as `@Unique` in some databases. The database primary keys are unique by definition. Some databases implement this by creating a unique index on the column.

Workaround

Do not specify both `@Id` and `@Unique` on a single field.

Cache Hit and Miss Counts May Rise Unexpectedly

Issue

Impacted Platforms: All

The cache hit and miss counts may rise unexpectedly when manipulating entities without the version data. The extra cache access occurs when EntityManager closes and all contained entities are detached. Entities without version fields appear to the system to be missing their version data, and the system responds by checking their version in the cache before detachment.

Workaround

Entities with version fields or other version strategies do not cause extra cache access.

Open JPA Tries to Create a Table Even if the Table Exists

Issue

Impacted Platforms: All

When using the MySQL database, if OpenJPA is configured to automatically run the mapping tool at runtime and create tables within the default schema, it will try to create the table even if the table exists in the database. For example:

```
<property name='openjpa.jdbc.SynchronizeMappings' value='buildSchema'/>
<property name='openjpa.jdbc.Schema' value='MySQL database name' />
```

A `PersistenceException` message will be thrown to indicate that the table already exists and the table creation statement fails.

Workaround

To avoid this problem, if you are using the MySQL database, do not configure OpenJPA to automatically run the mapping tool at runtime and specify the default schema at the same time.

EJB Applications Fail During Serialization

Issue

Impacted Platforms: All

EJB applications that use IIOP and send JPA entities from the server to the client will fail during deserialization if the entities are Serializable (but not Externalizable) and do not declare a `writeObject()` method.

Workaround

Add a `writeObject()` method to such entity classes. The write object can be trivial:

```
private void
writeObject(java.io.ObjectOutputStream out)
   throws IOException {
  out.defaultWriteObject();
}
```

EJB Handle Serialization Fails When Using IIOP to Invoke 11g EJB

Issue

Impacted Platforms: All

As described in the Enterprise JavaBeans Specification, the remote home interface allows a client to obtain a handle for that remote home interface. The handle can be serialized and written to stable storage. Later, possibly in a different JVM, the handle can be deserialized from stable storage and used to obtain a reference to the remote home interface.

However, when a servlet hosted on Oracle WebLogic Server 12.2.1 (and later) uses the IIOP protocol to look up the remote home interface of an EJB hosted on Oracle WebLogic Server 11g, serialize the handle to a file, and then deserialize the handle from the file, the deserialization may fail with a `ClassNotFoundException`.

The situation also applies to an EJB hosted on Oracle WebLogic Server 11g that attempts to use IIOP to invoke an EJB hosted on Oracle WebLogic Server 12.2.1 (and later).

Workaround

Use the T3 protocol instead of IIOP.

WebLogic Client Cannot Use IIOP to Look Up EJB 3.x Specified with the Remote Annotation

Issue

Impacted Platforms: All

When an application using the WebLogic standard client (`wlclient.jar`) uses the IIOP protocol to look up the remote home interface of an EJB 3.*x* bean that has been decorated with the `@Remote` annotation, the lookup may fail with the `ClassNotFoundException` message.

Workaround

You can do one of the following:

- In the client classpath, use the WebLogic Install client, `weblogic.jar`, instead of the WebLogic standard client, `wlclient.jar`.
- Create the remote home interface of the EJB by extending the `java.rmi.Remote` interface instead of using the `@Remote` annotation.

HTTP Publish/Subscribe Server Issues and Workarounds

This section describes the following issues and workarounds:

Authentication and Authorization of the Local Client is Not Supported

Issue

Impacted Platforms: All

The HTTP Publish/Subscribe server does not support authentication and authorization of the local client. The local client has full permission to operate on channels of the HTTP Publish/Subscribe server, which means that the local client can create/delete channels and publish/subscribe events from channels.

Workaround

N/A

Event Messages Published by Local Clients Do Not Go Through Filters

Issue

Impacted Platforms: All

Event messages published to a channel by a local client will not go through the message filters configured to that channel.

Workaround

N/A

Installation and Patching Issues and Workarounds

This section describes the following issues and workarounds:

Installation Fails with Fatal Error

Issue

Impacted Platforms: All UNIX

The installer does not verify whether sufficient disk space is available on the machine prior to completing the installation. As a result, if an installation cannot be completed due to insufficient space, the installer displays the following error message and exits:

```
Fatal error encountered during file installation. The installer will now
cleanup and exit!
```

Workaround

If this problem occurs, restart the installer using the following command:

```
server103_linux32.bin -log=log.out -log_priority=debug
```

The preceding command generates a log of the installation procedure, providing details about the exact cause of the failure. If the cause is indeed insufficient space, the log file indicates it explicitly.

JDBC Driver Fixes Are Not Included in the Installer for MAC OS X

Issue

Impacted Platforms: N/A

When installing Oracle WebLogic Server 12.2.1 on Mac OS X development systems, some recommended JDBC driver fixes for production environments are not included with the Oracle JDBC thin driver that is included with the installation.

Workaround

A patch or patch(es) will be made available on My Oracle Support for developers wishing to incorporate these fixes on development environments.

FAILED Server State Interrupts ZDT Rollout During the Shutdown Operation

Issue

Impacted Platforms: All

During server shutdown when executing rollouts, if the server enters the FAILED state, you may encounter the following error:

```
Workflow wf0008 failed and the revert process was not initiated. The failure was:
Failure performing execute of wf0008-3-1-0 (ShutdownServerResumeOnRevertCommand), caused
by Failed to shut down server server1: java.lang.Exception: The process for the server
server1 has not completely shut down. This problem should be reported and you may have
to kill the process manually.
```

This error is encountered when using WLST.

Workaround

You must check the server logs carefully to determine the reason for the server failure which led to the shutdown failure. You can continue the rollout after resolving the issue and manually shutting down the server.

ONS JAR Packaged with WebLogic Server 12.2.1.1 Requires Patch

Issue

Impacted Platforms: All

Due to incompatibility between the ONS (Oracle Notification Services) and the UCP (Universal Connection Pool) JARs that are packaged with WebLogic Server 12.2.1.1, the Active GridLink data source deployments are affected as some data sources stop receiving database FAN events.

Workaround

The ONS JAR packaged with WebLogic Server 12.2.1.1 requires a patch. You can contact My Oracle Support for the patch.

Long Wait for JMS T3 Standalone Clients During JMS Service Failback

Issue

Impacted Platforms: N/A

In the Zero Downtime (ZDT) patching scenario, when the rollout in progress involves JMS service migration with a case of failback, the JMS T3 standalone clients are unable to perform JNDI lookup for the connection factory and the destination objects on the JMS server.

Workaround

When the migrated service fails back to the original server, the JMS standalone clients must wait for a nondeterministic time until the JNDI lookup is successful.

Java EE Issues and Workarounds

This section describes the following issues and workarounds:

FastSwap May Relax the Access Modifiers of Fields and Methods

Issue

Impacted Platforms: All

FastSwap may relax the access modifiers of fields and methods. Private and protected members may be made public at runtime. This change alters the behavior of reflection and may affect reflection-based frameworks such as Struts.

Workaround

N/A

FastSwap Does Not Support Redefinition of the Entity Bean and ejbClass

Issue

Impacted Platforms: All

FastSwap does not support redefinition of the Entity bean and ejbClass (Session/MDB). Therefore, any updates to entity classes will cause redefinition errors.

Workaround

After updating an entity class, redeploy the application.

Classpath Order Is Not Guaranteed When There Are Multiple JARs in an EAR File

Issue

Impacted Platforms: All

When you have an EAR file containing separate JAR files, and two or more of those JAR files have a class with the same name, it is not possible to predict from which of those JAR files Oracle WebLogic Server will instantiate the class. This is not an issue if the classes are the same, but if they are different implementations, the results are unpredictable.

Workaround

Currently there is no known workaround for this issue.

FastSwap Not Supported When Using CDI

Issue

Impacted Platforms: All

FastSwap is not supported when using CDI. If you deploy an application in exploded format with FastSwap enabled, this deployment fails and errors related to CDI occur.

Workaround

N/A

JDK Issues and Workarounds

This section describes the following issues and workarounds:

Oracle JRockit Not Supported For Running WebLogic Server 12.1.2 and Later Server Applications

Issue

Impacted Platforms: All

Oracle WebLogic Server 12.1.2 supports JDK 7 for running server applications, and JDK 6 and JDK 7 for WebLogic Server 12.1.2 clients connecting to WebLogic Server 12.1.2 servers. Oracle JRockit is not supported for running WebLogic Server 12.1.2 and later server applications. For more information, see Supported Configuration in *What's New in Oracle WebLogic Server*.

Workaround

N/A

JMS Issues and Workarounds

This section describes the following issues and workarounds:

Change in Behavior of Unmapped Connection Factory Resources

Issue

Impacted Platforms: All

This issue may occur if you are using an EJB or servlet with resource reference to a JMS Connection Factory, that is, when a connection factory is obtained using an @Resource annotation or a context lookup of a resource defined in an application's XML descriptor, and if this resource reference does not explicitly specify a JNDI name via a lookup attribute, a mappedName attribute, or a jndi-name in a descriptor file.

In WebLogic Server 12.2.1 and later releases, the Java EE 7 Platform Specification mandates a change that can cause such resource references to unexpectedly return a Platform Default Connection Factory instead of either returning a connection factory from JNDI with a JNDI name that matches the resource name, or returning a `javax.naming.NameNotFoundException`. Following are some of the possible symptoms of this change in behavior of unmapped connection factory resources:

- An INFO level log message indicating that an application has been given a default connection factory though it might have required a custom connection factory instead. For example,

  ```
  BEA-169827> <The resource reference "jms/my_cf" of type JMS Connection Factory in
  application "my_module" does not specify a JNDI name. As of Java EE 7 and WebLogic
  version 12.2.1.1, such references return a "java:comp/DefaultJMSConnectionFactory"
  by default when no Connection Factory with a JNDI name that matches the resource
  name is found.>
  ```

- An application stops delivering customized behavior that is expected from a custom configured WL JMS connection factory, such as a default message expiration.
- Errors, such as `Illegal Destination type`, which may occur while attempting to use an AQ JMS destination with a WebLogic JMS connection factory.
- Exceptions such as `Destination not found` or `Dispatcher not found`.
- Temporary destination is created on a local JMS Server instead of a JMS Server that is hosted in the same cluster as the intended connection factory.

Workaround

To ensure that applications work as expected, we recommend one of the following workarounds:

- Explicitly configure the system to force old behavior by setting the WebLogic Server `JMSConnectionFactoryUnmappedResRefMode` configurable to the **FailSafe** mode. Note that this setting is compliant with the Java EE 7 specification although Java EE 7 mandates that this setting cannot default to the **FailSafe** mode. For more information, see Specifying the Unmapped Resource Reference Mode for Connection Factories in *Administering JMS Resources for Oracle WebLogic Server*.
- Examine and fix all unmapped resource reference in all servlets and EJB applications to make them explicitly specify their desired connection factory. If the desired connection factory is Java EE 7 default connection factory, then you can specify `java:comp/DefaultJMSConnectionFactory` as the JNDI name.

Deployment Descriptor Validation Fails

Issue

Impacted Platforms: All

Deployment descriptor validation fails when descriptor validation is enabled, and an EAR file contains only JMS modules.

Workaround

Ensure that there is at least one Java EE specification-compliant module in the EAR file.

Exception When Multiple Producers Use the Same Client SAF Instance

Issue

Impacted Platforms: All

When multiple JMS producers use the same JMS Client SAF instance (within a single JVM), depending on the timing of the JMS SAF client creation, you might receive the following exception:

```
Error getting GXA resource [Root exception is weblogic.jms.common.JMSException:
weblogic.messaging.kernel.KernelException: Error getting GXA resource]
```

Workaround

When using multiple JMS SAF client producers, introduce a small delay between the creation of each new client.

Multi-Byte Characters are Not Supported in Store File and Directory Names

Issue

Impacted Platforms: All

There is no support for multi-byte characters in the Oracle WebLogic Server Store file and directory names. For instance, if the Oracle WebLogic Server name has multi-byte characters, the default store is not created, and Oracle WebLogic Server fails to boot.

Workaround

Create the Oracle WebLogic Server instances without multi-byte characters in the path name and use that path name for the default store configuration. Do not use multi-byte characters.

Testing Abrupt Failures of WebLogic Server When Using File Stores on NFS

Issue

Impacted Platforms: All

Oracle strongly recommends verifying the behavior of a server restart after abrupt machine failures when the JMS messages and transaction logs are stored on an NFS mounted directory. Depending on the NFS implementation, different issues can arise post failover/ restart.

Workaround

None.

Custom Domain Template Upgrade May Result in Lost Topic Messages or Depleted Server Memory

Issue

Impacted Platforms: All

As of Oracle WebLogic Server 12.1.2, JMS server and WebLogic store targeting in the Configuration Wizard has changed.

In 12.1.2, the Configuration Wizard automatically targets JMS servers and WebLogic stores to migratable targets when these objects are not explicitly targeted to a Managed Server or a cluster in a domain template. Using migratable targets is a best practice that enables high availability for the JMS system.

If you use a custom domain template to create domains in Oracle WebLogic Server 12.1.2, and the template includes JMS servers and WebLogic stores that are not explicitly targeted to a Managed Server or a cluster, targeting results differ from previous releases.

This change in behavior also results in a change to durable topic subscriptions for message-driven beans (MDBs) that enable the `generate-unique-client-id` extension. When Oracle WebLogic Server creates durable topic subscriptions for such an MDB, it changes the subscription name to include the migratable target name. Messages stored under the original subscription names are not delivered to the MDB, and the original subscriptions continue to accumulate new messages.

When planning your upgrade, note the following important changes:

- Use the Reconfiguration Wizard to reconfigure your existing pre-12.1.2 domain for keeping the configuration and durable topic subscriptions intact. For instructions, see Reconfiguring a WebLogic Domain in *Upgrading Oracle WebLogic Server*.
- If you regenerate your domain using a custom template, as described above, the resulting configuration differs from previous releases and new durable topic subscriptions are created when the system is started. However, old durable topic subscriptions remain. Those subscriptions contain unprocessed messages that continue to accumulate messages, depleting the server memory.

Workaround

Choose one of the following recommended workarounds:

- Use the Reconfiguration Wizard to perform an in-place upgrade of the the domain.
- Drain messages before upgrading or regenerating the domain configuration.
- Delete the JMS file store files or JMS JDBC store tables. All messages persisted in the file or table are deleted.

Set System Properties for Interoperability with Existing JMS .NET Clients

Issue

Impacted Platforms: All

To enable JMS .NET clients developed prior to WebLogic Server 12.1.3 to interoperate with WebLogic Server 12.1.3, set the following system property on your WebLogic Server 12.1.3 instances:

```
-Dweblogic.protocol.t3.login.replyWithRel10Content=true
```

The default value is `false` for interoperability with existing JMS .NET clients developed prior to WebLogic Server 12.1.3.

Workaround

None.

JMS Distributed Destinations Are Not Present After Extending a Domain

Platform: All

Issue

Impacted Platforms: All

After extending a domain using an extension template that was generated from a domain that contains JMS distributed destinations, the distributed destinations are not present in the domain. This impacts the following distributed destinations:

- *distributed-queue*
- *distributed-topic*
- *uniform-distributed-queue*
- *uniform-distributed-topic*

If any of these elements are contained in the JMS XML files in the source template, they are not processed and are not configured in the destination domain.

Workaround

To resolve this, use the following sequence of WLST commands, either interactively or in a script:

```
readDomain('domain_path')

addTemplate('extension_template_file')

unassign('JmsSystemResource','resource_name','Target','destination_name')
For example: unassign('JmsSystemResource','JMSModule','Target','C1')

assign('JmsSystemResource','resource_name','Target','destination_name')
For example: assign('JmsSystemResource','testModule','Target','Server-1')

unassign('JmsSystemResource','resource_name','Target','destination_name')
For example: unassign('JmsSystemResource','testModule','Target','Server-1')

assign('JmsSystemResource','resource_name','Target','destination_name')For example:
assign('JmsSystemResource','testModule','Target','C1')

updateDomain()

closeDomain()
```

Using JBoss 5 as a Foreign Provider for a JMS Messaging Bridge Causes Issues

Issue

Impacted Platforms: All

When creating a JMS messaging bridge using JBoss 5 as a foreign provider, conflicting versions of the same class are loaded. This causes the bridge to fail during startup.

Workaround

Oracle recommends that you upgrade to JBoss 7 to avoid this issue.

JTA Issues and Workarounds

This section describes the following issue and workaround:

Transaction Protocol Changes May Cause Inconsistent Transaction Outcomes

Issue

Impacted Platforms: All

When the transaction protocol for a data source is changed, it is necessary to restart all targeted servers in order to avoid inconsistent transaction outcomes.

Workaround

N/A

Warning Message Repeatedly Logged on Enabling JDBC Store

Issue

Impacted Platforms: All

The following warning message is logged multiple times when JDBC TLog Store is used without enabling the Cross-Site recovery feature.

```
<Warning> <JTA> <BEA-111020> <An issue occurred during cross-site recovery
processing: PeerSiteRecoveryLeaseMaintenance: Unable to create either
connection or prepared statements for cross-site recovery..>
```

Workaround

You can work around this issue by using domain logging filter.

Commit Recovery Fails When Multiple DB2 Resources in a Transaction Point to the Same DB

Issue

Impacted Platforms: All

Commit recovery fails when multiple DB2 resources in a transaction point to the same DB.

Workaround

N/A

Java Virtual Machine (JVM) Issues and Workarounds

This section describes the following issues and workarounds:

- JVM Stack Overflow
- Using AWT libraries May Cause a JVM Crash
- Manually Set the Location of Java Endorsed Directory for Oracle WebLogic Server When Upgrading to 14.1.1.0.0 Running on Java SE 8

1.4 Thin Client Applet Cannot Contact Oracle WebLogic Server

Issue

Impacted Platforms: All

Due to a known Sun Microsystems VM bug (513552), a 1.4 Thin Client Applet cannot contact Oracle WebLogic Server 9.0 or later. This is because the VM does not distinguish correctly between a client and a server connection. The VM creates a server-type connection and caches it. It then attempts to make a client-type connection, finds the cached connection and tries to use that, but encounters an error because clients are not allowed to use server connections.

Workaround

N/A

Applications Running on Some Processors May Experience Intermittent Time Issues

Issue

Impacted Platforms: RedHat Linux

Applications that run on RedHat (RH) Linux and that also directly or indirectly use system time calls may experience intermittent time issues if the `ClockSource` is set to `tsc` (the default). The standard POSIX C `gettimeofday()` call, and consequently also the Java `System.currentTimeMillis()` and `java.util.Date()` calls can intermittently return a value that is approximately 4400 seconds in the future, even in a single-threaded application.

This issue is not unique to WebLogic or Java, but applies to any application running on RH Linux. Issues can occur for applications that either explicitly make a time call using standard Java, or explicitly by using any time-based application server services.

Possible symptoms include, but are not limited to, premature transaction timeouts, unexpected expiration of JMS messages, and incorrectly scheduled timers.

This issue was fixed in RedHat 5.3.

Workaround

N/A

JRockit JVM Appears to Freeze When Doing Long Array Copies

Issue

Impacted Platforms: Linux

The JRockit JVM appears to freeze when doing long array copies as part of unlimited forward rolling. This can happen when multiple server reboots occur due to `Out Of Memory` conditions.

Workaround

When booting the servers, include the following JRockit JVM flag:

```
-XXrollforwardretrylimit:-1
```

Serial Version UID Mismatch

Issue

Impacted Platforms: Linux

A Serial Version UID Mismatch issue is encountered if you deploy an application on a latest JVM, but compiled with previous Service Release of IBM Java 6 JDK.

Workaround

To be compatible with the serialization of previously compiled applications, modify the *WL_HOME*`/common/bin/commEnv.sh` file to include the following command:

```
JAVA_OPTIONS="$JAVA_OPTIONS
-Dcom.sun.xml.namespace.QName.useCompatibleSerialVersionUID=1.0"
```

Alternatively, you can use the command line option:

```
export IBM_JAVA_OPTIONS=
"-Dcom.sun.xml.namespace.QName.useCompatibleSerialVersionUID=1.0"
```

If you intend to deploy new applications with previously compiled applications, they must be recompiled as necessary to have the same Serial Version UID.

JVM Stack Overflow

Issue

Impacted Platforms: Linux

You might encounter a JVM stack overflow error or exception while running WebLogic Server. This issue applies to Oracle Enterprise Linux 4, 5, 5.1 on AMD64 and 64-bit Xeon platforms.

Workaround

Increase the stack size from the default 128k to 256k.

Using AWT libraries May Cause a JVM Crash

Issue

Impacted Platforms: Linux x86

You might encounter a JVM crash when using GUI libraries such as AWT or javax.swing (which often delegates to AWT).

Workaround

Start the server using the following flag:

```
-Djava.awt.headless=true
```

Manually Set the Location of Java Endorsed Directory for Oracle WebLogic Server When Upgrading to 14.1.1.0.0 Running on Java SE 8

Issue

Impacted Platforms: All

When upgrading an Oracle WebLogic Server 10.3.x domain to a domain running on Java SE 8, you may have to manually set the location of the Java endorsed directory (or directories) for Oracle WebLogic Server.

Workaround

You must manually set the location of the Java endorsed directory for Oracle WebLogic Server in the command you use to start the Managed Servers, if:

- You are using custom start scripts, that is, start scripts that are not provided by Oracle.
- You are trying to create an empty domain using `java.weblogic.Server`.

In any of these cases, include the `java.endorsed.dirs` parameter in the Managed Server startup command.

```
startWeblogic.sh -Djava.endorsed.dirs=ORACLE_HOME/oracle_common/modules/
endorsed
```

> **Note:**
>
> In all of the options described in this section, you must replace *ORACLE_HOME* with the absolute path to your Oracle WebLogic Server installation.

You can also specify this value when calling `startServer` by passing the values as `jvmArgs` or when calling `nmstart` by passing them as properties, such as:

```
wls:/nm/mydomain> prps =
          makePropertiesObject("Arguments=-
Djava.endorsed.dirs=ORACLE_HOME/oracle_common/modules/endorsed")wls:/nm/
mydomain> nmStart("AdminServer",props=prps)
```

If you are using Node Manager to start the Managed Server, you can include the `-Djava.endorsed.dirs=ORACLE_HOME/oracle_common/modules/endorsed` parameter in the ServerStartMBean's `arguments` attribute using WLST.

Life Cycle Management Issues and Workarounds

This section describes the following issue and workaround:

- Lifecycle Config Plug-In Path May Not be Updated After Unpacking a Domain

Lifecycle Config Plug-In Path May Not be Updated After Unpacking a Domain

Issue

Impacted Platforms: All

When packing a domain that has been configured to use Lifecycle Manager in one environment using the `pack` command, and then unpacking it in another environment, the plug-in paths contained in the `<Domain>/config/lifecycle-config.xml` file may still reference the Oracle Home directory paths from the source environment.

Workaround

Rectify the file paths in the `lifecycle-config.xml` file to reflect the Oracle Home directory paths in the target environment.

Monitoring Issues and Workarounds

This section describes the following issues and workarounds:

MBean Attributes Not Explicitly Marked as @unharvestable Appear as Harvestable

Issue

Impacted Platforms: All

The `@unharvestable` tag is not being honored at the interface level. If MBean attributes are not explicitly marked as `@unharvestable`, they are considered to be harvestable

Workaround

You can explicitly mark MBean attributes as `@unharvestable`.

Issue with Ambiguous Watch Rule ObjectName Patterns

Issue

Impacted Platforms: All

When specifying a wildcard pattern in a variable for a watch rule expression that matches custom MBean ObjectName patterns, ensure that the pattern is sufficiently explicit. If you exclude an MBean type name and use an ambiguous instance pattern, the following may result:

- Only WebLogic Server runtime MBean instances are matched to the pattern.
- The desired custom MBean instances are ignored.

For example, the following ObjectName pattern does not explicitly declare a type and uses an ambiguous ObjectName pattern that can match a WebLogic Server runtime MBean instance:

```
${ServerRuntime//com.b*:Type=Server*,*}
```

Workaround

To avoid confusion, use a sufficiently explicit ObjectName pattern, or declare the MBean type in the variable expression.

Behavior Change in CreateSystemResourceControl

Issue

Impacted Platforms: All

This issue is related to a change in how WebLogic Diagnostic Framework (WLDF) uses the module name for harvester records and watch rule notifications. The internal descriptor name is now overridden to use the name that is provided when the external WLDF descriptor is registered through the Runtime Control API or WLST functions. You will notice this if you have been using the Runtime Control feature to deploy external WLDF system resources to gather Harvester metrics, or listen for a Watch rule notification based on the deployed module.

For example, assume that the Harvester and Watch elements in your deployed descriptor resemble the following:

```
<harvester>
   <name>MyExternalResource</name>
...
<watch-notification>
   <name>MyExternalResource></name>
...
```

In addition, you register this descriptor with the runtime control as `createSystemControl("resource1", ...)`. Then the previous harvester data would have been recorded using `MyExternalResource` as the `WLDFMODULE` column value for Harvester records in the archive for this resource. It would also be used for the module name in the Watch Notification payloads. Now, `resource1` would be used for the `WLDFMODULE` name in the harvester records and the watch and notification payloads.

Workaround

Use the name the external WLDF resource was registered with when using the WLST command `createSystemResourceControl()`. Additionally, any notification listeners for Watch notifications from an external resource that are dependent on the WLDF module name in the notification payload should be looking for the name the control was registered with.

For example, if you register your control as `createSystemResourceControl("resource1", ...)`, then the WLDF Accessor queries for this resource should include the module name as `WLDFMODULE='resource1'` in the query string.

Node Manager Issues and Workarounds

This section describes the following Node Manager issues and workarounds:

- Removing the Primary Interface Causes Error During Server Migration
- Oracle HTTP Server Instances Start in the UNKNOWN State

Removing the Primary Interface Causes Error During Server Migration

Issue

Impacted Platforms: Linux

On some specific Linux platforms and versions, there is an issue removing a virtual interface/alias dynamically. Removing the virtual interface that is the primary address of the interface may result in other secondary virtual IP addresses being removed at the same time. This condition may lead to random exceptions occurring with Node Manager during server migration. If you encounter this issue, you may occasionally find exceptions in the Node Manager log file when shutting down a server after migration. For example, you may receive the following error:

```
java.io.IOException: Command '/<PATH to DOMAIN>/bin/server_migration/
wlsifconfig.sh -removeif -IPv4 eth0 X.X.X.X returned an unsuccessful exit code
'1'.
```

Here is an example of the issue:

First, add three virtual interfaces with the first one being the primary:

```
$ sudo /sbin/ifconfig eth0:4 X.X.X.178 netmask 255.255.248.0
$ sudo /sbin/ifconfig eth0:5 X.X.X.179 netmask 255.255.248.0
$ sudo /sbin/ifconfig eth0:6 X.X.X.180 netmask 255.255.248.0
$ sudo /sbin/ifconfig eth0:4 down
```

When removing the primary (the first interface in the list), the other two will be automatically removed at the same time.

Workaround

To fix this issue temporarily, use the following command to enable the `promote_secondaries` flag on your network interface. Replace `eth0` with your actual interface name:

```
$ sudo /sbin/sysctl net.ipv4.conf.eth0.promote_secondaries=1
```

You can also use the following command to update the default setting for all interfaces:

```
$ sudo /sbin/sysctl net.ipv4.conf.all.promote_secondaries=1
```

If this flag is enabled and the primary address of an interface gets deleted, a secondary interface will be upgraded to become the primary interface. The default is to purge all the secondary interfaces when you delete the primary interface.

To permanently remedy this issue after server reboot, update the `sysctl.conf` file. For example:

```
$ echo "net.ipv4.conf.eth0.promote_secondaries=1" >> /etc/sysctl.conf
```

Oracle HTTP Server Instances Start in the UNKNOWN State

Issue

Impacted Platforms: All

In rare cases, the Oracle HTTP Server (OHS) instances that are managed by the Oracle WebLogic Server may start in the `UNKNOWN` state. This situation can occur if the Administration Server is unable to initialize the state of the OHS instance. For example, if Node Manager is not running when you create the OHS instance and you connect directly to the Node Manager and bypass the Administration Server when checking the state for the first time.

Workaround

Continue to use the Administration Server. The state of the OHS instance should be initialized properly.

New Node Manager Property Names Cannot Be Used From WLST Offline

Issue

Impacted Platforms: All

WLST offline, as well as the `pack` and `unpack` commands, do not support setting the following new Node Manager replacement properties that were introduced in WebLogic Server 12.1.3.

Deprecated Property	Replacement Property
`CipherSuite`	`CipherSuites`
`CoherenceStartScriptEnabled`	`coherence.StartScriptEnabled`
`CoherenceStartScriptName`	`coherence.StartScriptName`
`IfConfigDir`	`weblogic.IfConfigDir`
`JavaHome`	Use `weblogic.startup.JavaHome` for WebLogic Server processes or `coherence.startup.JavaHome` for Coherence processes.
`StartScriptEnabled`	`weblogic.StartScriptEnabled`
`StartScriptName`	`weblogic.StartScriptName`
`StopScriptEnabled`	`weblogic.StopScriptEnabled`
`StopScriptName`	`weblogic.StopScriptName`
`UseMACBroadcast`	`weblogic.UseMACBroadcast`

Workaround

If you configure Node Manager properties using WLST offline, or the `pack` and `unpack` commands, you must continue to use the preceding deprecated properties, which remain fully supported in WebLogic Server. For more information, see Node Manager Properties in *Administering Node Manager for Oracle WebLogic Server*.

nmStart Fails to Start Administration Server for Partition Domain When Upgraded from Oracle WebLogic Sever 12.2.1.3.0 to 14.1.1.0.0

Issue

Impacted Platforms: All

As Oracle WebLogic Server 14.1.1.0.0 does not support partition, `nmStart` fails to start Administration Server for partition domain when upgraded from Oracle WebLogic Sever

12.2.1.3.0 to 14.1.1.0.0. Without partition, regular cluster domain upgrade or rollback works fine after you run the Reconfiguration Wizard.

Workaround

N/A

Operations, Administration, and Management Issues and Workarounds

There are no known Operations, Administration, and Management issues in this release of Oracle WebLogic Server.

Oracle Kodo Issues and Workarounds

This section describes the following Oracle Kodo issue and workaround:

- Value Retrieved for an Empty Byte Array Field is NULL

Value Retrieved for an Empty Byte Array Field is NULL

Issue

Impacted Platforms: MS Windows 2000

When trying to persist an empty byte array field within an entity to a Sybase or Oracle database, the value gets stored as a NULL rather than as bytes. As a result, when retrieving the value, NULL is returned.

This is a limitation of the Sybase and Oracle drivers, which convert the empty byte array to a NULL while storing it in the database. The issue happens with WebLogic JDBC drivers as well as the proprietary Sybase and Oracle drivers.

Workaround

N/A

Oracle WebLogic Server Proxy Plug-Ins Issues and Workarounds

This section describes the following issues and workarounds for the various Oracle WebLogic Server Proxy Plug-ins:

- GZIP Compression Does Not Work When HTTP/2 Protocol is Configured for a Back-End Connection
- No Error When Multiple HTTP2-Settings Header Fields are Sent During HTTP/2 Upgrade
- Inconsistent HTTP Error Codes are Returned for Unimplemented Methods
- Server Push Functionality is Not Supported When HTML Link Preload Option is Used
- Connection Error Reported Instead of Stream Error With WINDOW_UPDATE Size of Zero
- apr_socket_connection Exception Occurs When Using the Proxy Plug-In for IIS

GZIP Compression Does Not Work When HTTP/2 Protocol is Configured for a Back-End Connection

Issue

Impacted Platforms: Linux

Oracle WebLogic Server does not support GZIP compression over HTTP/2 connection. The request fails for applications that use GZIP compression to serve the request, in the following cases:

- When both the front-end and the back-end connections use HTTP/2 protocol.
- When the front-end connection uses HTTP/1.1 protocol and the back-end connection uses HTTP/2 protocol.

Workaround

N/A

No Error When Multiple HTTP2-Settings Header Fields are Sent During HTTP/2 Upgrade

Issue

Impacted Platforms: Linux

This issue occurs when using Oracle WebLogic Server 14.1.2.0.0 Proxy Plug-in for Apache HTTP Server.

A request that upgrades from HTTP/1.1 to HTTP/2 must include one `HTTP2-Settings` header field, and the server must not upgrade the connection to HTTP/2 if this header field is not present or if more than one is present. Also, the server must not send this header field. This is the expected behavior. However, when a client sends multiple `HTTP2-Settings` header fields to Apache HTTP Server when trying to upgrade the connection from HTTP/1.1 to HTTP/2, the Apache HTTP Server considers the first `HTTP2-Settings` frame and ignores the rest. The request to upgrade to HTTP/2 succeeds without any error.

Workaround

N/A

Inconsistent HTTP Error Codes are Returned for Unimplemented Methods

Issue

Impacted Platforms: Linux

This issue occurs when using Oracle WebLogic Server 14.1.2.0.0 Proxy Plug-in for Apache HTTP Server.

If you use any protocol other than HTTP/1.1, the Java Servlet 4.0 implementation used by Oracle WebLogic Server 14.1.1.0.0 returns a `400 (Bad Request)` response status code for unimplemented request methods, instead of returning a `405 (Method Not Allowed)` error. Consequently, an HTTP/2 request for an unimplemented method results in a `400` response status code.

Workaround

N/A

Server Push Functionality is Not Supported When HTML Link Preload Option is Used

Issue

Impacted Platforms: Linux

This issue occurs when using Oracle WebLogic Server 14.1.2.0.0 Proxy Plug-in for Apache HTTP Server.

The Server Push functionality is not supported when the HTML link header preload option is used as shown in the following example:

```
<head>
  <meta charset="utf-8">
  <title>JS and CSS preload example</title>

  <link rel="preload" href="style.css" as="style">
  <link rel="preload" href="main.js" as="script">

  <link rel="stylesheet" href="style.css">
  <script type="text/javascript" src="main.js"></script>
</head>
```

Workaround

Use one of the following options:

- Option 1: Set the response header in the servlet or JSP Java code on the Application Server side:

  ```
  response.setHeader("Link","style.css>; rel=preload; as=style,<main.js>;
  rel=preload; as=script");
  ```

- Option 2: Configure the required resources in the `httpd.conf` file:

  ```
  <Location />
    Header add Link "<style.css>;rel=preload"
    Header add Link "<main.js>;rel=preload"
  </Location>
  ```

Connection Error Reported Instead of Stream Error With WINDOW_UPDATE Size of Zero

Issue

Impacted Platforms: Linux

This issue occurs when using Oracle WebLogic Server 14.1.2.0.0 Proxy Plug-in for Apache HTTP Server.

As per the HTTP/2 specification, a receiver must treat the receipt of a `WINDOW_UPDATE` frame with a flow-control window increment of `0` (zero) as a stream error of type `PROTOCOL_ERROR`. Errors on the connection flow-control window must be treated as a connection error. However, when using the Oracle WebLogic Server 14.1.2.0.0 Proxy Plug-in for Apache HTTP Server, when a `WINDOW_UPDATE` frame with an increment of `0` is sent for a stream (stream ID other than `0`), it results in a connection error instead of a stream error, with Apache HTTP Server sending a `GOAWAY` frame instead of a `RST_STREAM` frame.

Workaround

NA

apr_socket_connection Exception Occurs When Using the Proxy Plug-In for IIS

Issue

Impacted Platforms: All

The proxy plug-in for IIS does not work in the following conditions and results in an `apr_socket_connection` error:

1. Both the IIS and Oracle WebLogic Server instances are on the same machine.
2. IPv6 is enabled on the machine, but the machine is not in an IPv6 environment (that is, the IPv6 interface is enabled but is not working).
3. The listen address of the Oracle WebLogic Server instance is set to the simple host name.
4. Either the directive WebLogicHost or WebLogicCluster is set to the simple host name for the IIS instance.

Workaround

N/A

Failure to Introspect Write Protected Domains With Managed Servers

Issue

Impacted Platforms: All

Introspection fails and users receive an error when they try to introspect a domain that they cannot write into.

Workaround

Change the permissions on the domain root directory to allow the user that executes the introspect.

SYSPROP Enables HTTP Proxying in OVAB Studio

Issue

Impacted Platforms: All

In Oracle Virtual Assembly Builder (OVAB) Studio, HTTP proxying is disabled. You can use a system property to enable HTTP proxy detection.

Workaround

You can set this system property for each execution of a Studio launch, or permanently by modifying the `abstudio.sh` file.

To set the property for a single execution of OVAB Studio:

1. Shut down OVAB Studio.
2. Remove the configuration directory:

   `$AB_INSTANCE/state/gui/$USER/system.12.1.2.0.0` (or equivalent)
3. Restart the GUI with the property set to some value, for example 1:

   ```
   ./abstudio.sh -J-Dovab.studio.enableHttpProxy=1
   ```

> **Note:**
>
> You must define the property in every ensuing execution of the GUI or the property setting in `abstudio.sh` will force proxying back to false.

To set the property to consistently enable HTTP proxying:

1. Edit the `abstudio.sh` file in the instance bin directory.
2. Add the property setting to SYSPROPS as follows:

   ```
   SYSPROPS="${SYSPROPS} -J-Dovab.studio.enableHttpProxy=1
   ```

After setting `enableHTTPProxy=1`, you can set the proxy host, port, and exceptions using the standard Java properties `http.proxyHost`, `http.proxyPort`, and `http.nonProxyHosts`. If you are using a non-standard desktop environment on Linux, you may need to set the `http_proxy` property with the `valuehost:port`.

RMI-IIOP Issues and Workarounds

This section describes the following issue and workaround:

Truncated Java Exception Stack Trace Returned to Client if EJB Invocation Fails

Issue

Impacted Platforms: All

When a client invokes an EJB that is hosted in a WebLogic 12.1.2 domain configured to run in production mode, any invocation failure results in a truncated Java exception stack trace returned to the client.

Workaround

In the Java command that starts WebLogic Server, specify the following option:

```
-Dweblogic.PrintStackTraceInProduction=true
```

Security Issues and Workarounds

This section describes the following issues and workarounds:

StoreBootIdentity Works Only if the Appropriate Server Security Directory Exists

Issue

Impacted Platforms: All

The `-Dweblogic.system.StoreBootIdentity` option works only if the appropriate server security directory exists. This directory is usually created by the Configuration Wizard or the upgrade tool.

However, the appropriate server security directory could be absent in domains checked into source-control systems.

Workaround

N/A

Boot Time Failure Occurs With SecurityServiceException

Issue

Impacted Platforms: All

A WebLogic Server instance can experience a boot time failure with a `SecurityServiceException` when the RDBMS Security Data Store is configured for a DB2 database using the DB2 driver supplied with WebLogic Server.

Workaround

When RDBMS Security Data Store is using the `AlternateId` connection property for a DB2 database, you must also set the additional property `BatchPerformanceWorkaround` as `true` when using the DB2 driver supplied with WebLogic Server.

InvalidParameterException Message Generated and Displayed

Issue

Impacted Platforms: All

After you configure either the Identity Provider or Service Provider services for SAML 2.0 and attempt to publish the SAML 2.0 services metadata file, an `InvalidParameterException` message may be generated.

Workaround

When configuring the SAML 2.0 federation services for a WebLogic Server instance, be sure to enable all binding types that are available for the SAML role being configured. For example, when configuring SAML 2.0 Identity Provider services, you should enable the POST, Redirect, and Artifact bindings. When configuring SAML 2.0 Service Provider services, enable the POST and Artifact bindings. Optionally, you may choose a preferred binding.

Enabling Both the Authentication and Passive Attributes In SAML 2.0 Service Provider Services Is an Invalid Configuration

Issue

Impacted Platforms: All

When configuring SAML 2.0 Service Provider services, enabling both the Force Authentication and Passive attributes is an invalid configuration that WebLogic Server is unable to detect. If both these attributes are enabled, and an unauthenticated user attempts to access a resource that is hosted at the Service Provider site, an exception is generated and the single sign-on session fails.

Even if the user is already authenticated at the Identity Provider site and Force Authentication is enabled, the user is forced to authenticate again at the Identity Provider site.

Avoid enabling both these attributes.

Workaround

N/A

Random Number Generator May Be Slow on Machines With Inadequate Entropy

Issue

Impacted Platforms: Linux

In order to generate random numbers that are not predictable, SSL security code relies upon "entropy" on a machine. Entropy is activity such as mouse movement, disk IO, or network traffic. If entropy is minimal or non-existent, then the random number generator will be slow, and security operations may time out. This may disrupt activities such as booting a Managed Server into a domain using a secure administrator channel. This issue generally occurs for a period after startup. Once sufficient entropy has been achieved on a JVM, the random number generator should be satisfied for the lifetime of the machine.

For further information, see Sun bugs 6202721 and 6521844 at:

http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=6202721

http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=6521844

Workaround

On low-entropy systems, you can use a non-blocking random number generator. To do this, add the `-Djava.security.egd=file:///dev/urandom` switch or `file://dev/./urandom` to the command that starts the Java process.

Optionally, you can raise the entropy level on Linux systems as follows:

- First, determine the current level

  ```
  cat/proc/sys/kernel/random/entropy_avail
  ```

- If the value is less than `500`, then you should configure the system to have more entropy.

  ```
  rngd -r /dev/urandom -o /dev/random -b -W 2048
  ```

  This will add bits to the entropy pool until the size is 2048.

Additional Information for BEA-090402 Message

Issue

Impacted Platforms: All

The `BEA-090402` message is a catalog message that explains what to do if a server instance fails to boot due to a problem with the `boot.properties` file.

However, the real issue is an authentication problem. `BEA-090402` describes the most likely root cause: the customer has modified the `boot.properties` file or the boot user password. Therefore, the authentication fails.

There are other causes for this failure that are less obvious. For instance, there could be an LDAP corruption, a disk failure, or a Managed Server may fail to connect to the Administration Server and falls back to authenticating on its local LDAP that is out-of-date. These causes are not mentioned in `BEA-090402`. If you are positive that you are not having a credential issue, `BEA-090402` may indicate one of these less common causes.

Workaround

N/A

LDAP Authenticator Log Messages Show Incorrect URL

Issue

Impacted Platforms: All

The Oracle WebLogic Server LDAP Authentication provider log messages show an incorrect URL for the LDAP connection returned by the `getConnection` method. The `getConnection` messages indicate that SSL is used, even if you did not specify SSL in the Oracle WebLogic Server provider configuration.

Workaround

The "Connecting to host" messages in the log file do correctly indicate whether SSL is used. For example,

- `<Connecting to host=somehost, port=3060>`
- `<Connecting to host=somehost, ssl port=3060>`

Security Errors Occur When Starting ODI Managed Server

Issue

Impacted Platforms: All

When starting an ODI Managed Server, the identity used to start the Managed Server is not mapping properly to the WebLogic Sever Administrator role, causing security errors. If these warnings are seen while starting the Managed Server, the embedded LDAP files from the Managed Server are causing a resynchronization of the policy from the Administration Server.

Workaround

Remove the embedded LDAP files in the folder `<domain>/…/data/ldap/ldapfiles` from the Managed Server and restart the Managed Server.

> **Note:**
>
> Do not delete the Embedded LDAP files from the Administration server as the Administration Server LDAP files contain the primary data for the security providers.

Changes in the Return Type of Methods in weblogic.security.net.ConnectionFilterImpl to Support the IPv6 Protocol

Impacted Platforms: All

Issue

In 14.1.1.0.0, the parse related methods of the `ConnectionFilterImpl` method return `BigInteger` instead of the `int` value. See Class ConnectionFilterImpl method in *Java API Reference for Oracle WebLogic Server*.

Workaround

N/A

Query String Removed from the Default HTTP Access Log Entries

Issue

Impacted Platforms: All

By default, the query string parameters may show up in the HTTP log files. This allows attackers access to sensitive data, such as passwords, personal information, database details, and so on. To overcome this vulnerability, we have removed the query string as part of the default access log entries.

To restore the query string to default HTTP access log entries or to previous behavior, set this system property `weblogic.servlet.access.log.default.format.with.query` to `TRUE`.

Workaround

N/A

Wrong Keystore Listed in AdminServer Log

Issue

Impacted Platforms: N/A

When starting the WLS Administration Server in secured production mode, the server logs refer to using a pkcs12 keystore file even though the server is actually using a `.jks` keystore file. Note that this is an error in the log message only (message ID `<BEA-090171>`); there is no runtime issue.

Workaround

N/A

Spring Framework on WebLogic Server Issues and Workarounds

This section describes the following issues and workarounds:

- OpenJPA ClassFileTranformer Does Not Work When Running on JRockit
- petclinic.ear Does Not Deploy on WebLogic Server

OpenJPA ClassFileTranformer Does Not Work When Running on JRockit

Issue

Impacted Platforms: All

The OpenJPA `ClassFileTranformer` does not work when running WebLogic Server on JRockit.

Workaround

Use an alternative method of applying enhancements at build time through an OpenJPA enhancer compiler; do not use the `LoadTimeWeaver`.

petclinic.ear Does Not Deploy on WebLogic Server

Issue

Impacted Platforms: All

For the SpringSource `petclinic` sample, the `petclinic.war` deploys without any problems. The `petclinic.ear` will not deploy on WebLogic Server because it is not packaged correctly. A request has been sent to SpringSource to fix the `petclinic.ear` packaging.

Workaround

N/A

Upgrade Issues and Workarounds

This section describes the following issues and workarounds:

SQLIntegrityConstraintViolationException May Occur When Upgrading

Issue

Impacted Platforms: All

The following exception occurs when upgrading Oracle WebLogic Server using the Reconfiguration Wizard with `log_priority=ALL`:

```
Internal Exception: java.sql.SQLIntegrityConstraintViolationException: ORA-00001: unique
constraint (NM_OPSS.IDX_JPS_RDN_PDN) violated
```

Workaround

N/A

Manually Configure DefaultIdentityAsserter During Domain Upgrade

Issue

Impacted Platforms: All

Several WebLogic Server software features, such as Lifecycle Manager, RESTful Management Services, WebLogic Remote Console, and Fusion Middleware Control have dependencies on WebLogic Server security features which may not be present in an upgraded domain configuration.

Workaround

You must manually configure the following security features during domain upgrade:

- Create an `LCMUser` service account if not present. This service account must be present in the administrator group and should have a secure password.
- Update the `DefaultIdentityAsserter` configuration to include `weblogic-jwt-token` as an active type if not present.

Web Applications Issues and Workarounds

This section describes the following issues and workarounds:

MaxPostSizeExceededException Reported in Web Browser

Issue

Impacted Platforms:All

After upgrading an application from a WebLogic Server version prior to 12.1.2, a `MaxPostSizeExceededException` is reported in the web browser.

Workaround

Set the `max-save-post-size` session-descriptor to the maximum size (in bytes) of the POST that will be saved or buffered by the container during FORM authentication.

Database Connections Become Unstable When a PoolLimitSQLException Error Occurs

Issue

Impacted Platforms: All

When a `PoolLimitSQLException` error occurs during a JDBC persistence session, connections to the database become unstable, and may fail with recovery or fail without recovery. This results in the loss of session data. Either an older session or null is returned.

Workaround

No workaround available.

Web Page Fails to Open When Accessing it Using the SSL Port

Issue

Impacted Platforms: All

When accessing a web page using the SSL port, the page fails to open and the following error is reported:

```
Secure Connection Failed

An error occurred during a connection to <hostname>.

You have received an invalid certificate. Please contact the server
administrator or email correspondent and give them the following information:

Your certificate contains the same serial number as another certificate
issued by the certificate authority. Please get a new certificate containing
a unique serial number.
```

Workaround

The following workaround can be used for Firefox.

If you have received this error and are trying to access a web page that has a self-signed certificate, perform the following steps in Firefox:

1. Go to **Tools** , select **Options**, click **Advanced**, and then select the **Encryption tab** and click **View Certificates**.
2. On the **Servers** tab, remove the certificates.
3. On the **Authorities** tab, find the Certificate Authority (CA) for the security device that is causing the issue and delete it.

If you are using Internet Explorer or other web browsers, you can ignore the warning page that appears and continue to the web page.

Deployment Plans Cannot be Used to Override Two Descriptors

Issue

Impacted Platforms: All

Deployment plans cannot be used to override the following two descriptors during the deployment of a web application or a web module: `WEB-INF/classes/META-INF/persistence.xml` and `WEB-INF/classes/META-INF/persistence-configuration.xml`. Deployment plans can otherwise be used to override any descriptor.

Workaround

Package `WEB-INF/classes/META-INF/persistence.xml` and `WEB-INF/classes/META-INF/persistence-configuration.xml` (if present) along with the related class files into a JAR file. The JAR file must then be placed in the `WEB-INF/lib` directory of the web application or web module. A deployment plan can be used to override the two descriptors in such a JAR file.

Spring Dependency Injection Not Supported on JSP Tag Handlers

Issue

Impacted Platforms: All

With the Spring extension model enabled, Oracle WebLogic Server 10.3 or later does not support Spring Dependency Injection (DI) on JSP tag handlers for performance reasons.

Currently, Oracle WebLogic Server supports Spring DI on most web components. For example, servlets, filters, and listeners. However, Spring DI is not presently supported on JSP tag handlers for performance reasons.

Workaround

No workaround available.

503 Error Occurs When Accessing an Application With a Valid sessionid

Issue

Impacted Platforms: All

When a session is persistent and an older version of a servlet context is retired, accessing the application with a valid `sessionid` will result in a 503 error.

For example, assume that the session-persistent type of a versioned web application is 'file'; a user can access the application successfully. Later, version 2 of the application is redeployed, and version 1 is retired. If the same user accesses the application, a 503 error occurs.

Workaround

No workaround available.

Applications Configuring jdbc-connection-timeout-secs Fail to Deploy

Issue

Impacted Platforms: All

As of Oracle WebLogic Server 12.1.2, the `jdbc-connection-timeout-secs` element in the `weblogic.xml` deployment descriptor has been removed. Applications that configure `jdbc-connection-timeout-secs` will fail to deploy on Oracle WebLogic Server 12.1.2 server instances, resulting in the following error in the server log:

```
Unable to load descriptor /.../WEB-INF/weblogic.xml of module myweb. The
error is weblogic.descriptor.DescriptorException: VALIDATION PROBLEMS WERE
FOUND
  <6:7> problem: cvc-complex-type.2.4a: Expected elements 'timeout-
secs@http://xmlns.oracle.com/weblogic/weblogic-web-app ...' instead of 'jdbc-
connection-timeout-secs@http://xmlns.oracle.com/weblogic/weblogic-web-app'
here in element session-descriptor@http://xmlns.oracle.com/weblogic/weblogic-
web-app
```

Workaround

Remove the `jdbc-connection-timeout-secs` element from the `weblogic.xml` deployment descriptor.

WebSocket: Server Cannot Receive Messages Larger Than 4MB

Issue

Impacted Platforms: All

In order to prevent incoming messages that are too large, Tyrus places a constraint on the message frame size. The default value is 4 MB.

Workaround

This value can be configured through the servlet context parameter. For WebLogic Server, this is the `weblogic.websocket.tyrus.incoming-buffer-size` parameter and it can be edited as follows:

```
<context-param>
   <param-name>weblogic.websocket.tyrus.incoming-buffer-size</param-name>
   <param-value>value_to_tune</param-value>
</context-param>
```

No Support for Annotations for JSF-Based Web Applications with Version Earlier than 2.5

Issue

Impacted Platforms: All

As of 12.2.1.1, Oracle WebLogic Server does not support annotated custom tags for JSF-based web applications that have the version of the web application deployment descriptor, `web.xml`, set to 2.5 or later.

Workaround

For compatibility with your web application that has the version set to 2.5 or earlier, Oracle recommends that you upgrade the version of the web application deployment descriptor, `web.xml`, to 3.2.

For instance, if your existing web application has the version set to 2.3, you should upgrade it to 3.2 as shown in the following example:

```
<?xml version = '1.0' encoding = 'windows-1252'?>
<web-app xmlns="xmlns.jcp.org/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="xmlns.jcp.org/xml/ns/javaee
xmlns.jcp.org/xml/ns/javaee/web-app_3_2.xsd"
         version="3.2">
   ...
</web-app>
```

WebLogic Server Scripting Tool (WLST) Issues and Workarounds

This section describes the following issues and workarounds:

- Behavior Changes in Jython Version 2.7.1
- Property Names Containing Certain Characters Are Not Supported by loadProperties
- Invalid cachedir Created by Jython Causes WLST to Error Out

Behavior Changes in Jython Version 2.7.1

The WLST scripting environment is based on the Java scripting interpreter, Jython. In Oracle WebLogic Server 14.1.1.0.0, the Jython version has been upgraded from version 2.2.1 to version 2.7.1. Jython 2.7.1 introduces new behaviors that may cause the WLST scripts to malfunction.

Note the following issues:

Jython Unicode string is prepended to each member in a string array

Jython 2.7.1 uses the default string type of Unicode. Existing strings will cause errors because they will be returned as `u'string'`. If the string value contains Unicode, then convert Unicode to ASCII by calling `str(x)`. In the following example, `host1` and `host2` have been prepended with `'u' string`, and `str(host)` has the value of `host1`:

```
java weblogic.WLST
connect('username', 'password', 't3://host:port')
cd('Servers/sysadmin/SingleSignOnServices/sysadmin')
ls()
wls:/domain/serverConfig/Servers/sysadmin/SingleSignOnService
s/sysadmin> cmo.getAllowedTargetHosts()

array(java.lang.String, [u'host1', u'host2'])
```

```
wls:/domain/serverConfig/Servers/sysadmin/SingleSignOnService
s/sysadmin> host = cmo.getAllowedTargetHosts()[1]
```

WebLogic Deploy Tooling support issue

Due to the Jython version upgrade in Oracle WebLogic Server 14.1.1.0.0, WebLogic Deploy Tooling 1.4.x and earlier do not support Oracle WebLogic Server 14.1.1.0.0.

Jython script upgrade failure

When using the latest Jython upgrade script, an error occurs if you create an empty ZIP file (`zipfile.py`). Close it and open it again for appending. The ZIP file fails to open and displays the `IllegalArgumentException` error.

`Javaos` replaced with `os`

The Python Lib module, `javaos`, has been replaced with the `os` module.

Change:

```
import javaos as os
```

To:

```
import os
```

Changes to the usage of the `main` value

The general PEP (Python Enhancement Proposal) standard for entering `main` (if the Python file is run as a script) is to check the name field to see if it is entered as a script, and then call the `main` function.

In Jython 2.2, the `__name__` field contains the value `main`. In the latest version, the `__name__` field contains the value `'__main__'`.

Change:

```
if __name__ == 'main':
    main(sys.argv)
```

To:

```
if __name__ == '__main__' or __name__ == 'main':
       main(sys.argv)
```

Modified Jython Script Functionality

Due to changes in functionality and bug fixes, the following changes to Jython scripts (*.py) are necessary:

- `raise` no longer accepts a string as input. So, you must create your own error type as shown in this example:

  ```
  raise Exception("my exception text")
  ```

- `ex.printStackTrace()` does not work. Instead, use the commands:

  ```
  import traceback
  traceback.print_exc()
  ```

- Some forms of the `import` function do not always work on JDK11 that Oracle WebLogic Server 14.1.1.0.0 supports. The following forms of the `import` function should work properly:

```
from some.package import *
import some.package.*

from some.package import SomeClass
import some.package.SomeClass
```

The explicit `from` form is most consistent as in this example:

```
from weblogic.coherence.descriptor.wl import
CoherenceClusterParamsBean
```

Jython 2.7.0 startup performance regression

See Jython Issue 2605.

`import` * does not work on JDK9 for `java`.*, `jdk`.* namespaces

See Jython Issue 2362.

Property Names Containing Certain Characters Are Not Supported by loadProperties

Issue

Impacted Platforms: All

The WLST `loadProperties` command does not support loading a property with a name that contains "." characters. For example, if the property `myapp.db.default` is present in the property file, WLST throws a name exception:

```
Problem invoking WLST - Traceback (innermost last):
  File "<iostream>", line 7, in ?
  File "<iostream>", line 4, in readCustomProperty
NameError: myapp
```

This is a system limitation of Python and the `loadProperties` command. WLST reads the variable names and values and sets them as variables in the Python interpreter. The Python interpreter uses "." as a delimiter to indicate module scoping for the namespace, or package naming, or both. Therefore, the properties file fails because `myapp.db.default.version=9i` is expected to be in the `myapp.db.default` package. This package does not exist.

Workaround

Use variable names that do not have periods. This will allow you to load the variables from the property file and refer to them in WLST scripts. You could use another character such as "_" or lowercase/uppercase character to delimit the namespace.

As an alternative, you can set variables from a properties files. When you use the variables in your script, during execution, the variables are replaced with the actual values from the properties file. For example:

```
myapp.py
var1=10
var2=20
import myapp
print myapp.var1
10
print myapp.var2
20
```

This will work for one level of namespaces (`myapp.var1`, `myapp.var2`). It will not work for top level variables that share the same name as the namespace (for example, `myapp=oracle` and `myapp.var1=10`). Setting the `myapp` variable will override the `myapp` namespace.

If you need multiple levels, then you can define a package namespace using directories. Create a myapp/db/default directory with a `vars.py` file as follows:

```
var1=10
var2=20
```

Then import:

```
import myapp.db.default.vars
print myapp.db.default.vars.var1
10
```

You may need to add `__init__.py` files to the subdirectories. Refer to the Python documentation for more information on packages:

https://docs.python.org/3/tutorial/index.html

Invalid cachedir Created by Jython Causes WLST to Error Out

Issue

Impacted Platforms: All

The default `cachedir` created by Jython 2.2 is not a valid directory. If you are using Jython directly from `weblogic.jar`, this causes WLST to error out.

Workaround

There are two workarounds for this issue:

- When invoking WLST, specify the `-Dpython.cachedir=<valid_directory>` parameter, or
- Install Jython 2.2.1 separately instead of using the partial Jython that is included in `weblogic.jar`.

Web Server Plug-Ins Issues and Workarounds

This section describes the following issue:

MOD_WLS_OHS Does Not Fail Over

Issue

Impacted Platforms: All

Currently, `mod_wl` and `mod_wl_ohs` only support container level failover and not application level failover. `mod_wl_ohs` continues to route requests to a down application as long as the managed server is up and running. In the clustered case, requests continue to go to the container where the original session started even when the application is shutdown, typically resulting in the http error 404.

Workaround

N/A

Web Services and XML Issues and Workarounds

This section describes the following issues and workarounds:

WSDL Compiler Does Not Generate Serializable Data Types

Issue

Impacted Platforms: All

The Web Service Description Language (WSDL) compiler does not generate serializable data types, so data cannot be passed to remote EJBs or stored in a JMS destination.

Workaround

N/A

Use of Custom Exception on a Callback

Issue

Impacted Platforms: All

Oracle WebLogic Server does not support using a custom exception on a callback that has a package that does not match the target namespace of the parent web service.

Workaround

Ensure that any custom exceptions that are used in callbacks are in a package that matches the target namespace of the parent web service.

Cannot Use JMS Transport in an Environment that Also Uses a Proxy Server

Issue

Impacted Platforms: All

You cannot use JMS transport in an environment that also uses a proxy server. The reason is that in the case of JMS transport, the web service client always uses the T3 protocol to connect to the web service, and proxy servers accept only HTTP/HTTPS.

Workaround

N/A

clientgen Fails When Processing a WSDL

Issue

Impacted Platforms: All

`clientgen` fails when processing a WSDL that uses the complex type `http://www.w3.org/2001/XMLSchema{schema}` as a web service parameter.

Workaround

N/A

IllegalArgumentException When Using a Two-Dimensional XML Object in a JWS Callback

Issue

Impacted Platforms: All

Using a two dimensional `XmlObject` parameter (`XmlObject[][]`) in a JWS callback produces an `IllegalArgumentException` error.

Workaround

No workaround available.

Using SoapElement[] Results in an Empty Array

Issue

Impacted Platforms: All

Using `SoapElement[]` as a web service parameter with `@WildcardBinding(className="javax.xml.soap.SOAPElement[]", binding=WildcardParticle.ANYTYPE)` will always result in an empty array on the client.

Workaround

Do not use the `@WildcardBinding` annotation to change the default binding of `SOAPElement[]` to `WildcardParticle.ANYTYPE`. The `SOAPElement[]` default binding is set to `WildcardParticle.ANY`.

FileNotFound Exception When a Web Service Invokes Another Web Service

Issue

Impacted Platforms: All

When web service 'A' wants to invoke web service 'B', web service 'A' should use the `@ServiceClient` annotation. If web service 'B' needs a custom policy file that is not attached to the WSDL for web service 'B', then web service 'A' will fail to run. Web service 'A' will look for the policy file at `/Web-Inf/classes/policies/`*`filename`*`.xml`. Since no policy file exists at that location, Oracle WebLogic Server will throw a 'file not found' exception.

Workaround

Attach the custom policy file to Web Service B, as in this example:

```
@Policy(uri="CustomPolicy.xml",
        attachToWsdl=true)
public class B {
  ...
}
```

Client Side Fails to Validate the Signature on the Server Response Message

Issue

Impacted Platforms: All

When the security policy has one of these Token Assertions, the client side may fail to validate the signature on the server response message.

```
  <sp:WssX509PkiPathV1Token11/>
  <sp:WssX509Pkcs7Token11/>
  <sp:WssX509PkiPathV1Token10/>
  <sp:WssX509Pkcs7Token10/>
```

In addition, when there are more than two certifications in the chain for X509 certification for `<sp:WssX509Pkcs7Token11/>` or `<sp:WssX509Pkcs7Token10/>` Token Assertion, the server side may fail to validate the signature on the incoming message.

A policy such as the following is not supported unless the entire certificate chain remains on the client side.

```
<sp:AsymmetricBinding>
   <wsp:Policy>
      <sp:InitiatorToken>
         <wsp:Policy>
            <sp:X509Token
               sp:IncludeToken='. . ./IncludeToken/AlwaysToRecipient'>
```

```
            <wsp:Policy>
               <sp:WssX509Pkcs7Token11/>
            </wsp:Policy>
         </sp:X509Token>
      </wsp:Policy>
      </sp:InitiatorToken>
      <sp:RecipientToken>
      <wsp:Policy>
      <sp:X509Token sp:IncludeToken='. . ./IncludeToken/Never'>
            <wsp:Policy>
               <sp:WssX509Pkcs7Token11/>
            </wsp:Policy>
         </sp:X509Token>
      </wsp:Policy>
      </sp:RecipientToken>
. . .
      </wsp:Policy>
   </sp:AsymmetricBinding>
```

Workaround

Use either of the following solutions:

1. Configure the response with the `<sp:WssX509V3Token10/>` Token Assertion, instead of `WssX509PkiPathV1Token11/>`. The policy will look like this:

```
<sp:AsymmetricBinding>
   <wsp:Policy>
     <sp:InitiatorToken>
        <wsp:Policy>
        <sp:X509Token sp:IncludeToken='. . ./IncludeToken/AlwaysToRecipient'>
           <wsp:Policy>
              WssX509PkiPathV1Token11/>
           </wsp:Policy>
        </sp:X509Token>
        </wsp:Policy>
     </sp:InitiatorToken>
     <sp:RecipientToken>
        <wsp:Policy> sp:IncludeToken='. . ./IncludeToken/Never'>
        <sp:X509Token
           <wsp:Policy>
              <sp:WssX509V3Token10/>
           </wsp:Policy>
        </sp:X509Token>
        </wsp:Policy>
     </sp:RecipientToken>
. . .
     </wsp:Policy>
   </sp:AsymmetricBinding>
```

2. Configure the response with the `WssX509PkiPathV1Token11/>` token assertion, but include it in the message. The policy will look like this:

```
 <sp:AsymmetricBinding>
   <wsp:Policy>
     <sp:InitiatorToken>
        <wsp:Policy>
        <sp:X509Token sp:IncludeToken='. . ./IncludeToken/AlwaysToRecipient'>
        <wsp:Policy>
           WssX509PkiPathV1Token11/>
        </wsp:Policy>
        </sp:X509Token>
     </wsp:Policy>
```

```
        </sp:InitiatorToken>
        <sp:RecipientToken>
           <wsp:Policy>
           <sp:X509Token sp:IncludeToken='. . ./IncludeToken/AlwaysToInitiator'>
              <wsp:Policy>
                 WssX509PkiPathV1Token11/>
               </wsp:Policy>
           </sp:X509Token>
           </wsp:Policy>
        </sp:RecipientToken>
    . . .
      </wsp:Policy>
    </sp:AsymmetricBinding>
```

When there are multiple certifications in the X509 Certificate chain, `<sp:WssX509PkiPathV1Token11/>` or `<sp:WssX509PkiPathV1Token10/>` should be used, instead of `<sp:WssX509Pkcs7Token11/>` or `<sp:WssX509Pkcs7Token10/>`.

xmlcatalog Element Entity Cannot be a Remote File or a File in an Archive

Issue

Impacted Platforms: All

For the `xmlcatalog` element in the `build.xml` file, the location of an entity must be a file on the local file system. It cannot be a remote file (for example, `http:`) or a file in an archive (for example, `jar:`).

Workaround

If necessary, define the remote element as an entity in a catalog file.

Local xmlcatalog Element Does Not Work Well

Issue

Impacted Platforms: All

The local `xmlcatalog` element does not work well due to an Ant limitation.

Workaround

In the Ant `build.xml` file, you have to define a local element above a `clientgen(wsdlc)` task when you are in the same target, or define the element outside of a target.

External Catalog File Cannot be Used in the xmlcatalog Element of clientgen

Issue

Impacted Platforms: All

An external catalog file cannot be used in the `xmlcatalog` element of a `clientgen` task. For example, this snippet of an Ant build file will not work:

```
<clientgen ...
  <xmlcatalog>
    <catalogpath>
```

```
    <pathelement location='wsdlcatalog.xml'/>
  </catalogpath>
</xmlcatalog>
```

This is a limitation of the Ant XML Catalog.

Workaround

Resource locations can be specified either inline or in an external catalog file(s), or both. To use an external catalog file, the `xml-commons` resolver library (`resolver.jar`) must be in the classpath. External catalog files may be either plain text format or XML format. If the `xml-commons` resolver library is not found in the classpath, external catalog files, specified in `<catalogpath>` paths, will be ignored and a warning will be logged. However, in this case, processing of inline entries will proceed normally.

Currently, only `<dtd>` and `<entity>` elements may be specified inline. These correspond to the OASIS catalog entry types PUBLIC and URI, respectively.

WS-AT Interoperation Issues With WebSphere and WebLogic Server

Issue

Impacted Platforms: All

Web Services Atomic Transactions (WS-AT) 1.1 interoperation using WebSphere as the client and either WebLogic Server or JRF as the service does not work.

WS-AT 1.1 interoperation does work when WebSphere is the service and either WebLogic Server or JRF is the client. In this case, interoperation works only if you have WebSphere 7 with Fix/Feature Pack 7.

Workaround

N/A

Occasional JAX-RS Error Message in the Server Log

Issue

Impacted Platforms: All

When shutting down an Oracle WebLogic Server instance that has a JAX-RS application running on it, you may observe the following error message in the server log:

```
<Error>
<org.glassfish.jersey.server.internal.monitoring.MonitoringStatisticsProcessor
> <BEA-000000> <Exception thrown when provider class
org.glassfish.jersey.server.internal.monitoring.MonitoringFeature$StatisticsLi
stener was processing MonitoringStatistics. Removing provider from further
processing. ...
```

Workaround

This error message is benign and can be ignored.

WebLogic Tuxedo Connector Issues and Workarounds

This section describes the following issue and workaround:

View Classes are not Set on a Per Connection Basis

Issue

Impacted Platforms: All

View classes are not set on a per connection basis.

A shared WebLogic Tuxedo Connector hash table can cause unexpected behavior in the server if two applications point to the same VIEW name with different definitions. There should be a hash table for the view classes on the connection as well as for the Resource section.

Workaround

Ensure that all VIEW classes defined across all your WebLogic Workshop applications are consistent, meaning that you have the same VIEW name representing the same VIEW class.