Oracle® Fusion Middleware

Release Notes for Oracle WebLogic Server

12*c* (12.2.1)

E55177-05

March 2016

This document describes all known issues for Oracle WebLogic Server 12.2.1.

ORACLE®

Oracle Fusion Middleware Release Notes for Oracle WebLogic Server, 12*c* (12.2.1)

E55177-05

Contents

Preface

This document describes all known issues with Oracle WebLogic Server 12*c* (12.2.1).

Audience

This document is intended for all users of Oracle WebLogic Server 12*c* (12.2.1).

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 customers that have purchased support have access to electronic support through My Oracle Support. For information, visit
`http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit
`http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Related Documents

For more information, see the following Oracle resources:

- Oracle WebLogic Server Online Documentation library

- Oracle Technology Network at
  `http://www.oracle.com/technetwork/index.html`.

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.
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.

x

1

Introduction

This chapter introduces Oracle Fusion Middleware Release Notes for Oracle WebLogic Server 12*c* (12.2.1).

This chapter includes the following sections:

- Section 1.1, "Latest Release Information"
- Section 1.2, "Purpose of This Document"
- Section 1.3, "System Requirements and Specifications"
- Section 1.4, "Certification Information"
- Section 1.5, "Licensing Information"

1.1 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 Technology Network at:

http://www.oracle.com/technetwork/indexes/documentation/index.html

1.2 Purpose of This Document

This document contains the release information for Oracle WebLogic Server 12*c* (12.2.1). It describes differences between the software and its documented functionality.

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

1.3 System Requirements and Specifications

Oracle WebLogic Server installation and configuration will not complete successfully unless users meet the hardware and software pre-requisite requirements before installation.

For more information, see the *Installing and Configuring Oracle WebLogic Server and Coherence*.

1.4 Certification Information

For certification information, see the Oracle Fusion Middleware Supported System Configurations page on Oracle Technology Network.

1.5 Licensing Information

Licensing information for Oracle WebLogic Server is available at:

http://shop.oracle.com

2

Oracle WebLogic Server

This chapter describes issues associated with Oracle WebLogic Server 12.2.1.

> **Note:** As of WebLogic Server 12.2.1, all information about deprecated and removed functionality in WebLogic Server is documented in *What's New in Oracle WebLogic Server 12.2.1* in the following sections:
>
> - "Deprecated Functionality (WebLogic Server 12*c*)"
> - "Removed Functionality and Components"

This chapter includes the following sections:

- General Issues and Workarounds
- Administration Console Issues and Workarounds
- Apache Beehive Support Issues and Workarounds
- Clustering Issues and Workarounds
- Configuration Issues and Workarounds
- Connector (Resource Adapter) Issues and Workarounds
- Console Extensions Issues and Workarounds
- Core Server and Core Work Manager Issues and Workarounds
- Data Source Issues and Workarounds
- Dependency Injection Issues and Workarounds
- Deployment Issues and Workarounds
- Developer Experience Issues and Workarounds
- EJB Issues and Workarounds
- Examples Issues and Workarounds
- HTTP Publish/Subscribe Server Issues and Workarounds
- Installation and Patching Issues and Workarounds
- Java EE Issues and Workarounds
- JDK Issues and Workarounds
- JMS Issues and Workarounds

- JNDI Issues and Workarounds

- JTA Issues and Workarounds

- Java Virtual Machine (JVM) Issues and Workarounds

- Life Cycle Management Issues and Workarounds

- Monitoring Issues and Workarounds

- Node Manager Issues and Workarounds

- Operations, Administration, and Management Issues and Workarounds

- Oracle Kodo Issues and Workarounds

- Plug-ins Issues and Workarounds

- Protocols Issues and Workarounds

- RMI-IIOP Issues and Workarounds

- Security Issues and Workarounds

- SNMP Issues and Workarounds

- Spring Framework on WebLogic Server Issues and Workarounds

- System Component Architecture (SCA) Issues and Workarounds

- Upgrade Issues and Workarounds

- Web Applications Issues and Workarounds

- WebLogic Server Scripting Tool (WLST) Issues and Workarounds

- Web Server Plug-Ins Issues and Workarounds

- Web Services and XML Issues and Workarounds

- WebLogic Tuxedo Connector Issues and Workarounds

- Documentation Errata

2.1 General Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.1.1, "Multi-Byte Characters Display Incorrectly in Filenames When Using Safari"

- Section 2.1.2, "Strong Password Enforcement May Cause Issues With WLST Offline Scripts"

- Section 2.1.3, "In Turkish Locale, MDS Initialization Fails"

- Section 2.1.4, "Administration Server Reports a 'Too Many Open Files' Message on the EM Console"

- Section 2.1.5, "Installation Requirements if Using Coherence With Maven"

- Section 2.1.6, "Default WebLogic Server Message Prefix Will Change"

- Section 2.1.7, "Utility Packages are Incorrectly Published"

2.1.1 Multi-Byte Characters Display Incorrectly in Filenames When Using Safari

Platform: 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()
```

2.1.2 Strong Password Enforcement May Cause Issues With WLST Offline Scripts

Platform: All

With the implementation of strong password enforcement (8 character minimum with one numeric or special character) in this release of WebLogic Server, existing scripts could potentially encounter issues.

Workaround

Use either of the following workarounds to bypass the new password restrictions.

- Set the `BACKWARD_COMPAT_PW_CHECK` environment variable to `true`.
- Include the `-Dbackward.compat.pw.check=true` option when invoking WLST.

Oracle recommends that you change passwords to comply with the new password requirements, as this variable and option will be removed in a future release of WebLogic Server.

2.1.3 In Turkish Locale, MDS Initialization Fails

Platform: All

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

Workaround

Start the server in English locale.

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

Platform: 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
   ```

2.1.5 Installation Requirements if Using Coherence With Maven

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

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

2.1.6 Default WebLogic Server Message Prefix Will Change

Platform: All

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

2.1.7 Utility Packages are Incorrectly Published

The utility Apache packages, `com.bea.util.jam` and `com.bea.util.jam.visitor` are incorrectly published in the Java API Reference document as being open-source.

These packages are not intended for public consumption.

2.2 Administration Console Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.2.1, "Pressing Browser Back Button Discards Context"

- Section 2.2.2, "Unsupported Work Manager Configurations Can Be Created"

- Section 2.2.3, "Server Status Table Reflects Inconsistent Information"

- Section 2.2.4, "Exceptions When Defining a Security Policy for an EJB"

- Section 2.2.5, "Administration Console Does Not Always Reflect External Changes Made in a Deployment Plan"

- Section 2.2.6, "Application Testing Links Fail to Resolve in Administration Console"

- Section 2.2.7, "java.lang.NoClassDefFoundError is Displayed"

- Section 2.2.8, "Error When Configuring Security Role For Newly Created Coherence Cluster Service or Cache"
- Section 2.2.9, "Online Help States "Start Node Manager Using a Shortcut on the Start Menu""
- Section 2.2.10, "Administration Console Extensibility is Deprecated"
- Section 2.2.11, "Inconsistent Behavior Between Consoles When Creating a Partition"

2.2.1 Pressing Browser Back Button Discards Context

Platform: All

After a page flow completes in the WebLogic Server Administration Console, it forwards to a different page, typically a table.

Pressing the browser Back button at this point results in an attempt to load the last JSP file in the completed assistant. At this point, all of the context for this assistant is discarded.

Workaround

Oracle recommends that you do not use the browser Back button to step back into an assistant once changes are cancelled or finished, and that you do not go back to a previous step in an assistant. Instead, use the navigation links and buttons in the WebLogic Server Administration Console.

2.2.2 Unsupported Work Manager Configurations Can Be Created

Platform: All

The WebLogic Server Administration Console permits the creation of Work Manager configurations that are not supported and do not function as intended. Incorrect Work Manager configurations may result in a number of exceptions being recorded in the server logs, most commonly `'Validation problems were found'` exceptions while parsing deployment descriptors.

Workaround

Follow the guidelines described in the online help for Work Manager configurations. Specifically, you can only assign one request class to any given Work Manager, and that request class must be of the same or a broader scope than the Work Manager. You should not assign an application-scoped request class to a global Work Manager, and you should not create more than one application-scoped request class for an application-scoped Work Manager.

Correcting the Work Manager configurations to match the documented constraints resolves these issues.

2.2.3 Server Status Table Reflects Inconsistent Information

Platform: All

The Server Status table on the **Cluster: Monitoring: Summary** page includes two default columns: **Primary** and **Secondary Distribution Names**. These fields do not always reflect all of the replication statistics that are collected and displayed on the **Cluster: Monitoring: Failover** page, depending on the replication scenario.

Please refer to the **Cluster: Monitoring: Failover** page for definitive information.

2.2.4 Exceptions When Defining a Security Policy for an EJB

Platform: All

When defining security policies in the WebLogic Server Administration Console for an EJB deployment that references types defined in a separate library deployment, exceptions can be observed if that library deployment is not available to the Console.

Workaround

All library deployments should be targeted at the WebLogic Server Administration Server as well as any Managed Servers needed to support referencing applications. This will ensure that when defining policies, the Console will have access to those library deployments so that referenced types can be class-loaded as needed.

2.2.5 Administration Console Does Not Always Reflect External Changes Made in a Deployment Plan

Platform: All

The WebLogic Server Administration Console does not always reflect external changes made in a deployment plan. If a change is made in a deployment plan outside of the Console (for example, using Workshop, editing the plan text files directly, or updating a deployment with a new plan using WLST or webLogic.Deployer) while a Console user is also viewing that deployment plan, the Console user will not see those changes.

Workaround

Navigate to a configuration page for a different deployment, then navigate back to the original deployment again.

2.2.6 Application Testing Links Fail to Resolve in Administration Console

Platform: All

In some configurations, the Application Testing pages included in the WebLogic Server Administration Console use IPv6 addresses in the testing links. These addresses are valid for WebLogic server instances, but in some mixed IPv4 and IPv6 environments, these addresses cannot be used from the browser to interact with applications and the testing links cannot be resolved.

Workaround

This scenario typically happens when an administrator does not specify the listen address for a server in the configuration and the server is running on a dual stack (IPv6/IPv4) machine where Java and the operating system are configured to use IPv6 in preference to IPv4. In these mixed environments where the IPv4 stack cannot communicate with IPv6, Oracle recommends starting all server instances with the following command so that all servers are downgraded to use IPv4 only:

```
-Djava.net.preferIPv4Stack=true
```

2.2.7 java.lang.NoClassDefFoundError is Displayed

Platform: All

While using the WebLogic Server Administration Console with applications or EJBs deployed on a Managed Server that depend on a deployed library, you may encounter a `java.lang.NoClassDefFoundError`.

Workaround

The WebLogic Server Administration Console needs access to any shared library deployments so that Java data types and annotations can be processed. Therefore, all shared library deployments should always be targeted to the WebLogic Server Administration Server in addition to any Managed Servers or clusters.

2.2.8 Error When Configuring Security Role For Newly Created Coherence Cluster Service or Cache

Platform: All

An unexpected error condition is noted in the WebLogic Server Administration Console when configuring a security role for a newly created Coherence cluster service or cache. It is a common pattern in the WebLogic Server Administration Console that newly created artifacts must be saved and activated before it is possible to access them to configure security roles and policies on those artifacts. Many console pages check this and display a message indicating that "This page is not available because the necessary security providers have not been configured, or those configuration changes are pending and not yet activated. Please activate the changes and (if necessary) restart the Administration Server to make this page available." This check is not present in the Coherence security pages.

Workaround

After creating a new Coherence cluster, activate the configuration changes and restart any servers as indicated in the restarts changelist. This ensures that the Coherence cluster resources are available for role and policy configuration.

2.2.9 Online Help States "Start Node Manager Using a Shortcut on the Start Menu"

Platform: MS Windows

The option to start Node Manager on Windows machines using a shortcut on the Start menu has been removed in WebLogic Server 12.1.2.

Workaround

Ignore this text: "On Windows, you can start Node Manager using a shortcut on the Start menu."

Use other methods to start Node Manager. See "Starting and Stopping Node Manager" in *Administering Node Manager for Oracle WebLogic Server*.

2.2.10 Administration Console Extensibility is Deprecated

Platform: All

Console extensibility is deprecated as of WebLogic Server 12.1.3. The related documentation, *Extending the Administration Console for Oracle WebLogic Server* and the *Java API Reference for the Oracle WebLogic Server Administration Console*, have been removed from the documentation set.

2.2.11 Inconsistent Behavior Between Consoles When Creating a Partition

Platform: All

Partitions created in the WebLogic Server Administration Console that do not contain a virtual target and resource group are not displayed on the Partition Monitoring page

in Fusion Middleware Control (FMWC). These partitions are displayed in the FMWC partitions table with an unknown status.

Workaround

As a workaround, select the partitions in the FMWC partitions table and add a virtual target and a resource group.

2.3 Apache Beehive Support Issues and Workarounds

There are no known Apache Beehive Support issues in this release of WebLogic Server.

2.4 Clustering Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.4.1, "Threads Are Blocked on Cluster Messaging in Unicast Mode"
- Section 2.4.2, "Impact of Minimum and Maximum Dynamic Cluster Size Constraints"

2.4.1 Threads Are Blocked on Cluster Messaging in Unicast Mode

Platform: 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.

2.4.2 Impact of Minimum and Maximum Dynamic Cluster Size Constraints

Platform: All

To support elasticity, 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.

2.5 Configuration Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.5.1, "ASProvWorkflowException Occurs When Creating a WebLogic Domain"
- Section 2.5.2, "Use the -Dfile.encoding Property When Running WLST in a Non-English Locale"
- Section 2.5.3, "Configuration Tools Can Fail If WebLogic Installation Path Contains Spaces"

2.5.1 ASProvWorkflowException Occurs When Creating a WebLogic Domain

Platform: 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 use these JAVA_OPTIONS, the applications 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 other application.

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

Platform: 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 via a DOS command window. This changes the default character set for the Java process. For example:

- The active code page for a DOS window is 850. This 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.

In this situation, you can start WLST as follows:

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

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

2.5.3 Configuration Tools Can Fail If WebLogic Installation Path Contains Spaces

Platform: MS Windows

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

Workaround

You must enable short name generation in the Windows registry in order for spaces to be properly handled 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.

2.5.4 Directory For a Non-Existent Server Name Is Created

Platform: 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.

2.5.5 Abnormal Behavior in Terminal Window After Entering WebLogic Password

Platform: 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.

2.5.6 Creating and Updating Domains Takes Too Long

Platform: Linux

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

- Installing WebLogic Server on UNIX or Linux operating systems if the Server Examples are included in the installation.
- Using the 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
```

2.5.7 Password Field Is Not Editable When Configuring a New Domain

Platform: Linux

On Linux systems, when creating a new domain in 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 that appears, 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 file `~/.scim/config` 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.

2.5.8 Administration Server Memory Consumption and JMX Notifications

Platform: 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 WebLogic Server level, to simulate the unregister 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 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.

The `managed-server-notifications-enabled` attribute can be set using WLST as follows:

```
edit()
startEdit()
cd("JMX/domain-name")
cmo.setManagedServerNotificationsEnabled(false)
activate()
```

2.5.9 Issue Rolling Back Changes For `editCustom()` MBeans

Platform: All

The `editCustom()` tree contains MBeans for upper stack and system component products. If you make changes to these MBeans, the changes are persisted immediately to the pending directory. This is different from the WebLogic Server MBeans in the `edit()` tree, which require an explicit save.

If you use `stopEdit()`, `cancelEdit()` or exit WLST with an open edit session, then the unsaved changes to the WebLogic Server MBeans will be rolled back. However, the changes to the `editCustom()` tree will not be rolled back since they have been persisted.

Workaround

Use the `undo('y')` command to rollback the unactivated changes to the `editCustom()` MBeans.

2.5.10 Issue Starting Multiple Development Servers On Same Host

Platform: All

Two development servers cannot be started on the same host if they are sharing a single Derby instance.

Workaround

Configure each development server to use its own unique instance of Derby instead of sharing a single Derby instance. For more information see "Running Each Domain with a Unique Derby Instance" in *Developer's Guide for Oracle Service Bus*.

2.5.11 Coherence Cache Override Not Working

Platform: All

If the 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 WebLogic Server cluster that is also created by the Configuration Wizard. If no Managed Server or 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 WebLogic Server configuration tool, which results in the Coherence cache configuration override file not being detected by the Coherence cluster. Please note that this issue only occurs if you are using the cache override feature.

Workaround

Use the following workaround:

1. Start the domain created with the Configuration Wizard and connect using the WebLogic Server Administration Console.

2. In the left pane of the WebLogic Server Administration Console, expand **Environment** and select **Coherence Clusters**.

3. Select your Coherence cluster. The Coherence cluster settings page is displayed.

4. Select the **Members** tab, which displays all of the members of the Coherence cluster.

5. Deselect the servers and clusters that are members of the Coherence cluster and click **Save**.

6. Reselect the servers and clusters that are the desired members for the Coherence cluster and click **Save**.

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.

2.5.12 Creating Managed Server Domain from a Template Causes Error

Platform: 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.

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

Platform: 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:

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

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

2.5.14 1st arg can't be coerced to String Error Occurs in WLST

Platform: All

When executing a WLST script or WLST command, a `TypeError: state(): 1st arg can't be coerced to String error occurs`. This error occurs because a WLST class name `server` was 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.

2.5.15 Partition Listener May Not Receive MBean Unregistration Notification

In some cases, a partition listener may not receive notifications when an MBean is unregistered from an MBean server, even though the partition is enabled to receive notifications. This is due to some limitation in the current implementation of WebLogic Server 12.2.1.

Workaround

Currently there is no known workaround for this issue.

2.5.16 WL MT Partition Scoped Configuration Changes Are Ignored

Changes to an existing RG and RGT scoped non-module configurations may be ignored even after a partition restart. Changes to resource group overrides may also be ignored. However, changes to module-scoped configurations such as data source tuning, JMS destinations, and connection factories are unaffected by this issue.

The changes appear to be correct if queried via WLST but old values continue to be consumed within the server instance.

Workaround

Restart the server for the new values to take effect.

2.5.17 Resource Group May Not Reflect Changes After Retargeting

This issue occurs when there are two resource group templates containing resources of the same type and sharing the same name.

When a resource group is initially targeted to one resource group template, and is then targeted to the second resource group template, and changes are then made to the resource with the shared name in the second resource group template, those changes may not be reflected. The resource group might continue to use the value from the resource in the first resource group template.

Workaround

To bypass this issue, use either of the following workarounds:

- Avoid using resources with the same name across multiple resource group templates.

- After making changes to the resource in the second resource group template, restart the server for the new changes to take effect.

2.5.18 No Support for Explicit Port and Port Offset for Global Virtual Targets

WebLogic Server provides support for setting the Explicit port and port offset for virtual targets. However, currently the setting for Explicit port and port offset can take effect only for virtual targets that are assigned to a partition. For Global Virtual Targets (the virtual targets that are not assigned to any partition), the settings for Explicit port and port offset do not take effect.

2.5.19 Limitation of Using the JMX Thin Client

Due to recent changes in JDK, WebLogic Server no longer supports JMX with just the `wlclient.jar` file. To use JMX, you must use either the WebLogic full client, `(weblogic.jar)`, or the `wljmxclient.jar` file.

2.6 Connector (Resource Adapter) Issues and Workarounds

This section describes the following issues and workarounds:

2.7 Console Extensions Issues and Workarounds

There are no known Extensions issues in this release of WebLogic Server.

2.8 Core Server and Core Work Manager Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.8.1, "Threads Become Stuck While Waiting to Get a Connection"

- Section 2.8.2, "Using IPv6-Formatted Addresses"

- Section 2.8.3, "Server Cannot Be Started After a Whole Server Migration"

- Section 2.8.4, "Object State is not Retained After Renaming Field"

- Section 2.8.5, "Servers Configured to Listen on a Host Name Are Listening on a Different Host Name After Startup"

- Section 2.8.6, "Administration Server or Node Manager Cannot Track the Status of a Managed Server"

- Section 2.8.7, "Multicast Traffic Observed to be Unreliable During or After a Network Partition"

- Section 2.8.8, "No Java DB Leasing Script or Support"

- Section 2.8.9, "Initial Connection May Remain Open When Using t3 Protocol with External Load Balancers"

- Section 2.8.10, "Managed Server Fails to Start When Added to Multicast Cluster"

- Section 2.8.11, "High Number of Application Threads May Cause a Server to Stall"

- Section 2.8.12, "Enterprise Beans Can be Accessed by Remote Clients When Annotated with @Local"

2.8.1 Threads Become Stuck While Waiting to Get a Connection

Platform: All

When a machine that is hosting one of the Managed Servers is abruptly shut down, a network cable is pulled, or its network interface card has issues, and any server attempts communication with that managed server, threads become stuck waiting to get a connection.

Workaround

This can currently be resolved by using a private flag:

```
-Dweblogic.client.SocketConnectTimeoutInSecs
```

and setting an appropriate timeout value that will release the thread attempting to make the connection and allow the request to fail quickly.

For more information about timeout properties, see "Setting Client Timeouts" in *Developing RMI Applications for Oracle WebLogic Server*.

2.8.2 Using IPv6-Formatted Addresses

Platform: All

When using an IPv6-formatted address for 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
```

2.8.3 Server Cannot Be Started After a Whole Server Migration

Platform: All

If the WebLogic Server Administration Server is down when a Whole Server Migration occurs for a clustered server, and the server migrates to a machine on which it was never run before, the server cannot be started on the new machine.

Workaround

Use one of the following workarounds for this issue:

- Ensure that the Administration Server is up when the server migration is being performed.

- Use a shared disk/NFS for all the migratable servers in the cluster.

2.8.4 Object State is not Retained After Renaming Field

Platform: 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 that when a field name is changed, it is treated as follows:

- the field with old name is deleted

- the field with new name is added

Thus, 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 reset.

Workaround

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

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

Platform: 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.

2.8.6 Administration Server or Node Manager Cannot Track the Status of a Managed Server

Platform: Linux

If you start a managed server by providing an incorrect WebLogic Server Administration Server URL from the command line (that is, the Administration Server cannot be reachable at the provided URL), the managed server will start in Managed Server Independence (MSI) mode.

In this case, neither the Administration Server nor Node Manager can track the status of the managed server. The WebLogic Server Administration Console will show the status of the managed server as UNKNOWN, but the server will actually be RUNNING in MSI mode.

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

Platform: 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.

2.8.8 No Java DB Leasing Script or Support

Platform: All

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.

There is no workaround for this issue.

2.8.9 Initial Connection May Remain Open When Using t3 Protocol with External Load Balancers

Platform: All

When using the t3 protocol with external load balancers, the initial connection may remain associated with the IP address of the load balancer used to do the bootstrap into the cluster. Therefore, a small percentage of requests may pass through the load balancer after the initial connection. This behavior, if present, is a side-effect of the implementation that should not be relied upon since Oracle reserves the right to change this behavior at any point in time.

2.8.10 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.

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"
```

2.8.11 High Number of Application Threads May Cause a Server to Stall

In some cases, a server JVM may stall and a thread dump of the stalled JVM may reveal that almost all the `weblogic.kernel.Default` threads appear to be stalled in calls like `wait-for-data` or `wait-for-prepare-acks`.

Workaround

Use one of the following workarounds to bypass this issue:

- Set the JTA Min Threads Constraint limit of 20 or 30 on all servers in all involved clusters for the JTACoordinatorWM work manager. You can do this by using the `-Dweblogic.transaction.jta.coordinator.wm.min.constraint=YYY` system property.

- If the previous workaround doesn't work, 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 number of threads in the thread dump, try 100, then 150, and so on. However, setting too many threads may cause very poor performance or even an `out of memory` error. You must also note that setting `MinPoolSize` to a value of 400 or above will require you to set `-Dweblogic.threadpool.MaxPoolSize` to the same value, failing which the value of `MinPoolSize` is ignored.

- Tune applications to use fewer threads. For examples related to thread management for Message-Driven Beans (MDB), see Tuning Message-Driven Beans in *Tuning Performance of Oracle WebLogic Server*.

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

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

2.9 Data Source Issues and Workarounds

This section describes the following issues and workarounds:

■ Section 2.9.1, "Data Source Validation Fails For SQL Server When Specifying a Port Number"

2.9.1 Data Source Validation Fails For SQL Server When Specifying a Port Number

Included per RN bug 20394122. no base bug. Added in 12.2.1. Check back next release.

Starting in WebLogic Server 12.2.1, when connecting 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

The JDBC descriptor file must either be updated to set the `allowPortWithNamedInstance` property to `true`, or the port number must be removed from the URL.

When a port number is specified, the WebLogic Server Administration Console automatically appends the URL with the `allowPortWithNamedInstance=true` property. The WLST scripts must be updated by the administrator to set this property to true. The following sample URLs are valid:

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

2.10 Dependency Injection Issues and Workarounds

This section describes the following issues and workarounds:

■ Section 2.10.1, "BeanManager Does Not Contain a ValidatorFactory Bean in Lifecycle Events"

■ Section 2.10.2, "CDI Enabled EAR with Non-CDI Enabled WAR Does Not Work Correctly"

■ Section 2.10.3, "CDI Treatment of RAR Changes in 12.2.1"

2.10.1 BeanManager Does Not Contain a ValidatorFactory Bean in Lifecycle Events

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.

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

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.

2.10.3 CDI Treatment of RAR Changes in 12.2.1

For CDI 1.0 in 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 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.

2.11 Deployment Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.11.1, "security-permission Element is not Available in weblogic-application.xml"

- Section 2.11.2, "Extraneous String Values Interpreted as File Specification"

- Section 2.11.3, "The restore Method Does Not Update the DConfig Bean With Plan Overrides"

- Section 2.11.4, "Deployment Task Fails When a Large Application File Is Deployed"

- Section 2.11.5, "Attempting to Redeploy an Application Fails if the Application is Already Deployed Using a Different Source File Location"

- Section 2.11.6, "Relevant Output Message Not Displayed"

■ Section 2.11.7, "Application Names Must Be Unique Within a Domain"

2.11.1 security-permission Element is not Available in weblogic-application.xml

Platform: 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 only apply security policies to JAR files that are EJBs or Web applications.

2.11.2 Extraneous String Values Interpreted as File Specification

Platform: All

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

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

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

2.11.3 The restore Method Does Not Update the DConfig Bean With Plan Overrides

Platform: All

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

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

the plan does not correctly override the DConfig Bean.

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();
```

2.11.4 Deployment Task Fails When a Large Application File Is Deployed

Platform: 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 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.

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

Platform: 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 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.

2.11.6 Relevant Output Message Not Displayed

Platform: 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.

2.11.7 Application Names Must Be Unique Within a Domain

Platform: All

As of WebLogic Server 12.2.1, when deploying an application to a domain with a specified application name, if that application name is already in use in the current domain, the application deployment fails.

This is a behavior change from previous WebLogic Server versions, when specifying the same application name caused WebLogic Server to automatically derive a unique name based on the specified name.

Workaround

To avoid application deployment failure, specify unique names for all applications in a domain.

2.12 Developer Experience Issues and Workarounds

This section describes the following issue and workaround:

- Section 2.12.1, "Users Need to Set BEA_HOME System Property While Using Appc For Pub-Sub Modules"

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

Platform: All

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

Workaround

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.

2.13 EJB Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.13.1, "Primary Key in Oracle Table is CHAR"
- Section 2.13.2, "No Available Annotation That Enables Creation of a Clusterable Timer"
- Section 2.13.3, "Kodo's MappingTool Cannot Generate Schemas"
- Section 2.13.4, "Extensions to the JPA Metadata Model Can Only Be Specified Via Annotations"
- Section 2.13.5, "Lookup Method Injection Not Supported by Spring"
- Section 2.13.6, "Deserializing a JDO PersistenceManagerFactory in a Managed Environment May Fail"
- Section 2.13.7, "Indexes Not Always Created During Schema Creation"
- Section 2.13.8, "OpenJPA throws an exception when @Id fields are also annotated as @Unique"
- Section 2.13.9, "Cache Hit and Miss Counts May Rise Unexpectedly"
- Section 2.13.10, "Open JPA Tries to Create a Table Even if the Table Exists"
- Section 2.13.11, "EJB Applications Fail During Serialization"
- Section 2.13.12, "Non-Transactional Message-Driven Bean Container Can Fail to Provide Reproducible Behavior For Foreign Topics"

2.13.1 Primary Key in Oracle Table is CHAR

Platform: 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.

2.13.2 No Available Annotation That Enables Creation of a Clusterable Timer

Platform: 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 put the `<timer-implementation>` element and corresponding values into the file.

2.13.3 Kodo's MappingTool Cannot Generate Schemas

Platform: 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.

2.13.4 Extensions to the JPA Metadata Model Can Only Be Specified Via Annotations

Platform: All

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

Workaround

To specify Kodo-specific metadata for your object model, either:

■ use the Kodo-specific annotations, or

■ convert your XML-based metadata to the JDO metadata format, which does support XML specification of extensions.

2.13.5 Lookup Method Injection Not Supported by Spring

Platform: All

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

2.13.6 Deserializing a JDO PersistenceManagerFactory in a Managed Environment May Fail

Platform: 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 a `PersistenceManagerFactory` should not generally be necessary in a managed environment.

2.13.7 Indexes Not Always Created During Schema Creation

Platform: 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.

2.13.8 OpenJPA throws an exception when @Id fields are also annotated as @Unique

Platform: All

OpenJPA throws an exception when `@Id` fields are also annotated as `@Unique` in some databases. 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.

2.13.9 Cache Hit and Miss Counts May Rise Unexpectedly

Platform: All

The cache hit and miss counts may rise unexpectedly when manipulating entities without version data. The extra cache access occurs when the 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.

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

Platform: All

When using the MySQL database, and OpenJPA is configured to automatically run the mapping tool at runtime and create tables within the default schema (for example):

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

OpenJPA will try to create the table even if the table already exists in the database. A PersistenceException 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, don't configure OpenJPA to automatically run the mapping tool at runtime and specify the default schema at the same time.

2.13.11 EJB Applications Fail During Serialization

Platform: 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();
```

```
        }
```

2.13.12 Non-Transactional Message-Driven Bean Container Can Fail to Provide Reproducible Behavior For Foreign Topics

Platform: All

When using multi-threaded processing for non-transactional topic Message-Driven Beans (MDBs) that specify a foreign topic (non-WebLogic) JMS, the MDB container can fail to provide reproducible behavior. For example, if a `runtimeException` is thrown in the `onmessage()` method, the container may still acknowledge the message.

Workaround

Set the `max-beans-in-free-pool` attribute to `1` in the deployment descriptor.

2.14 Examples Issues and Workarounds

There are no known Examples issues in this release of WebLogic Server.

2.15 HTTP Publish/Subscribe Server Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.15.1, "Authentication and Authorization of the Local Client is not Supported"
- Section 2.15.2, "Event Messages Published By Local Clients Do Not Go Through Filters"

2.15.1 Authentication and Authorization of the Local Client is not Supported

Platform: All

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

2.15.2 Event Messages Published By Local Clients Do Not Go Through Filters

Platform: All

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

2.16 Installation and Patching Issues and Workarounds

This section describes the following issue and workaround:

- Section 2.16.1, "Installation Fails with Fatal Error"
- Section 2.16.2, "rollingRestart Command Causes Intermittent Session Loss for EJB Clients"
- Section 2.16.3, "JDBC Driver Fixes Are Not Included in the Installer for MAC OS X"

- Section 2.16.4, "FAILED Server State Interrupts ZDT Rollout During Shutdown Operation"

- Section 2.16.5, "ONS Jar Packaged with WebLogic Server 12.2.1 Requires Patch"

- Section 2.16.6, "Long Wait for JMS T3 Standalone Clients during JMS Service Failback"

2.16.1 Installation Fails with Fatal Error

Platform: 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.

2.16.2 rollingRestart Command Causes Intermittent Session Loss for EJB Clients

Platform: All

When using the `rollingRestart` WLST command to sequentially restart server instances in a domain, a client of an EJB application that is deployed on a cluster in that domain may experience intermittent session loss.

This behavior may occur in a domain that contains a cluster that includes multiple Managed Servers hosted on one machine under the following circumstances:

- The EJB is a stateful session bean.

- The client has continuous access to the EJB using the T3 protocol.

- The `rollingRestart` command is entered.

Workaround

If your EJB application is using stateful session bean, then it recommended that you do not configure two Managed Servers on the same machine. For instance, if a cluster contains three Managed Servers, it is recommended to have each Managed Server configured on a different machine.

Set up the replication group such that the Managed Server on one machine uses another Managed Server (located on a machine different from the one that hosts the primary session state) as the backup server for replication. For more information about replication group, see Failover and Replication in a Cluster in *Oracle Fusion Middleware Administering Clusters for Oracle WebLogic Server*.

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

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.

2.16.4 FAILED Server State Interrupts ZDT Rollout During Shutdown Operation

During the 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 or the Administration Console.

Workaround

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

2.16.5 ONS Jar Packaged with WebLogic Server 12.2.1 Requires Patch

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, 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 requires a patch. You can contact My Oracle Support for the patch.

2.16.6 Long Wait for JMS T3 Standalone Clients during JMS Service Failback

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 look up for the connection factory and destination objects on the JMS server.

Workaround

When the migrated service fails back to the original server, the JMS standalone clients must wait for nondeterministic time till the JNDI looked up successfully.

2.17 Java EE Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.17.1, "FastSwap May Relax the Access Modifiers of Fields and Methods"

- Section 2.17.2, "FastSwap Does Not Support Redefinition of the Entity Bean and ejbClass"
- Section 2.17.3, "Classpath Order Is Not Guaranteed When There Are Multiple JARs in an EAR File"
- Section 2.17.4, "FastSwap Not Supported When Using CDI"
- Section 2.17.5, "No Library is Loaded if the library-directory Element is Empty"
- Section 2.17.6, "Disable Implicit CDI Scanning When Migrating Java EE 6 Applications to Java EE 7"

2.17.1 FastSwap May Relax the Access Modifiers of Fields and Methods

Platform: All

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

2.17.2 FastSwap Does Not Support Redefinition of the Entity Bean and ejbClass

Platform: 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.

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

Platform: 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 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.

2.17.4 FastSwap Not Supported When Using CDI

Platform: 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.

2.17.5 No Library is Loaded if the library-directory Element is Empty

Platform: All

The Java EE 7 specification (https://jcp.org/en/jsr/detail?id=342) states that an empty library-directory element may be used to specify that there is no library directory.

In previous releases of WebLogic Server, an empty `library-directory` element in an `application.xml` file, as shown below, resulted in the library JAR files in the root directory of an EAR file being loaded as libraries:

```
<library-directory></library-directory>
```

As of WebLogic Server 12*c* (12.2.1) this behavior was changed to adhere to the Java EE 7 specification. Now, if an empty `library-directory` element exists in the `application.xml` file, no library is loaded from the application root, the `lib` directory, or a shared library.

> **Note:** If you create an empty directory in the application and point the `library-directory` element to that directory, you can load libraries from shared libraries (but not libraries from the `lib` directory) without removing libraries from the `lib` directory.

2.17.6 Disable Implicit CDI Scanning When Migrating Java EE 6 Applications to Java EE 7

When migrating existing applications to WebLogic Server 12.2.1, the deployment performance of applications developed using Java EE 6 (or prior Java EE versions) may be adversely affected by changes to default CDI scanning required by Java EE 7.

Workaround

To improve deployment performance for applications developed on previous Java EE versions, you can disable implicit CDI scanning in Java EE 7 by setting `implicit-bean-discovery-enabled` to `false` by:

- Using the WebLogic Server Administration console, go to the Domain General Configuration page, in the Advanced Setting Sections, and deselect **Enable Implicit Bean Discovery**.

- Using WLST online or offline.

  - To change this setting using WLST online, enter the following commands:

    ```
    connect('user','password','url')
    domainConfig()
    edit()
    cd('CdiContainer/mydomain')
    startEdit()
    set('ImplicitBeanDiscoveryEnabled',0)  // 1 to enable 0 to disable
    validate()
    save()
    activate(block="true")
    ```

  - To change this setting using WLST offline, enter the following commands:

    ```
    readDomain(domain)
    create('mydomain','CdiContainer')
    cd('CdiContainer/mydomain')
    set('ImplicitBeanDiscoveryEnabled',0)
    // 1 to enable 0 to disable
    updateDomain()
    closeDomain()
    ```

2.18 JDK Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.18.1, "Oracle JRockit Not Supported For Execution of WebLogic Server 12.1.2 and Later Server Applications"

2.18.1 Oracle JRockit Not Supported For Execution of WebLogic Server 12.1.2 and Later Server Applications

Platform: All

Oracle WebLogic Server 12.1.2 supports JDK 7 for execution of 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 execution of WebLogic Server 12.1.2 and later server applications. For more information, see "Supported Configuration" in *What's New in Oracle WebLogic Server 12.2.1*.

2.19 JMS Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.19.1, "Change in Behavior of Unmapped Connection Factory Resources"
- Section 2.19.2, "Deployment Descriptor Validation Fails"
- Section 2.19.3, "Exception When Multiple Producers Use the Same Client SAF Instance"
- Section 2.19.4, "Multi-byte Characters are not Supported in Store File and Directory Names"
- Section 2.19.5, "Testing Abrupt Failures of WebLogic Server When Using File Stores on NFS"
- Section 2.19.6, "Custom Domain Template Upgrade May Result in Lost Topic Messages or Deplete Server Memory"
- Section 2.19.7, "Set System Properties for Interoperability with Existing JMS .NET Clients"
- Section 2.19.8, "JMS Distributed Destinations Are Not Present After Extending a Domain"
- Section 2.19.9, "Using JBoss 5 as a Foreign Provider for a JMS Messaging Bridge Causes Issues"
- Section 2.19.10, "Memory Leak When Frequently Creating JMS Temporary Destinations"
- Section 2.19.11, "Administration Console Monitoring Fails to Show Migrated JMS Server Instances"

2.19.1 Change in Behavior of Unmapped Connection Factory Resources

Platform: 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, 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.

2.19.2 Deployment Descriptor Validation Fails

Platform: All

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

Workaround

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

2.19.3 Exception When Multiple Producers Use the Same Client SAF Instance

Platform: 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, try introducing a small delay between the creation of each new client.

2.19.4 Multi-byte Characters are not Supported in Store File and Directory Names

Platform: All

There is no support for multi-byte characters in WebLogic Store file and directory names. For instance, when the WebLogic Server name has multi-byte characters, the default store cannot be created, and WebLogic Server will not boot.

Workaround

Create 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 in the WebLogic Server name.

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

Platform: 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. For more information, see Section 6.3, "Testing Abrupt Failures of WebLogic Server When Using File Stores on NFS."

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

Platform: All

As of 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 WebLogic Server 12.1.2, and that template includes JMS servers and WebLogic stores that are not explicitly targeted to a Managed Server or a cluster, targeting results will 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 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 will not be delivered to the MDB, and the original subscriptions will continue to accumulate new messages.

When planning your upgrade, note the following important changes:

- If you follow the instructions in *Upgrading Oracle WebLogic Server* and use the Reconfiguration Wizard to reconfigure your existing pre-12.1.2 domain, the configuration and durable topic subscriptions will remain intact.

- If you regenerate your domain using a custom template, as described above, the resulting configuration will differ from previous releases and new durable topic subscriptions will be created when the system is started. However, old durable topic subscriptions will remain. Those subscriptions may contain unprocessed messages that will continue to accumulate messages, depleting server memory.

Workaround

Choose one of the following recommended workarounds:

- Use the Reconfiguration Wizard to upgrade the domain in place.

- Drain messages before upgrading or regenerating the domain configuration. After upgrading, use the WebLogic Server Administration Console to search and delete old JMS subscriptions.

- Delete JMS file store files or JMS JDBC store tables. Note that all messages persisted in the file or table will be deleted.

2.19.7 Set System Properties for Interoperability with Existing JMS .NET Clients

Platform: 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.

2.19.8 JMS Distributed Destinations Are Not Present After Extending a Domain

Platform: 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()
```

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

Platform: 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.

2.19.10 Memory Leak When Frequently Creating JMS Temporary Destinations

Platform: All

Frequently creating JMS temporary destinations may lead to slow performance, stuck threads, or JVM failure due to memory leak. This may impact messaging applications that directly create temporary destinations. This could also affect applications that use Web Services Async beans as the WS-Async framework may frequently create temporary destinations internally.

To verify whether this issue is the cause of a memory leak, check if a JVM heap dump reveals that there are far more `weblogic.jms.backend.BEQueueRuntimeDelegate` objects than concurrent client threads.

Workaround

You can either contact My Oracle Support for a patch, or modify applications to cache and reuse temporary destinations.

2.19.11 Administration Console Monitoring Fails to Show Migrated JMS Server Instances

The WebLogic Server Administration Console fails to show monitoring information for JMS Server instances that have failed over from their preferred server to another server in the cluster. This issue occurs only with cluster-targeted JMS Servers that have High Availability enabled by referencing a store configured with the `RestartOnFailure` or `Always` option as its migration policy.

Workaround

Use an alternate monitoring approach such as WLST.

2.20 JNDI Issues and Workarounds

There are no known JNDI issues in this release of WebLogic Server.

2.21 JTA Issues and Workarounds

This section describes the following issue and its workaround:

- Section 2.21.1, "Unexpected Migration of JTA When Using Automatic Migration Failure Recovery Services"

- Section 2.21.2, "Transaction Protocol Changes May Cause Inconsistent Transaction Outcomes"

2.21.1 Unexpected Migration of JTA When Using Automatic Migration Failure Recovery Services

If the service migration policy is set to **Auto-Migrate Failure-Recovery Services**, and an administrator forcefully shuts down the User Preferred Server (UPS), the JTA Transaction Recovery Service may migrate to another server in the cluster. This migration may occur *only* if the user has set the blocking attributes, `weblogic.transaction.blocking.commit=true` and `weblogic.transaction.blocking.rollback=true`, to retry 2PC calls infinitely, and if the time taken for these retires to complete exceeds the forceful shutdown timeout. However, this is in contrast to the expected behavior wherein the service should *not* migrate if an administrator shuts down the UPS (either gracefully or forcibly).

Possible symptoms of this issue include, unexpected early completion of pending transactions, and unexpected invocation of pre-migration and post-migration scripts (if such scripts are configured).

For more information about the automatic migration of the failure recovery services, see "Service Migration" in *Administering Clusters for Oracle WebLogic Server*.

Workaround

Use one of the following workarounds to bypass this issue:

- Set the `CompletionTimeoutSeconds` attribute to a value that retries fewer number of 2PC calls.

- Set the `ServerLifeCycleTimeoutVal` attribute (server's default forceful shutdown timeout) to a value exceeding the value set for `CompletionTimeoutSeconds`.

2.21.2 Transaction Protocol Changes May Cause Inconsistent Transaction Outcomes

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.

2.22 Java Virtual Machine (JVM) Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.22.1, "1.4 Thin Client Applet Cannot Contact WebLogic Server"

- Section 2.22.2, "Applications Running on Some Processors May Experience Intermittent Time Issues"

- Section 2.22.3, "JRockit JVM Appears to Freeze When Doing Long Array Copies"

2.22.1 1.4 Thin Client Applet Cannot Contact WebLogic Server

Platform: All

Due to a known Sun Microsystems VM bug (513552), a 1.4 Thin Client Applet cannot contact 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 then encounters an error because clients are not allowed to use server connections.

2.22.2 Applications Running on Some Processors May Experience Intermittent Time Issues

Platform: 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.

2.22.3 JRockit JVM Appears to Freeze When Doing Long Array Copies

Platform: 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
```

2.22.4 Serial Version UID Mismatch

Platform: 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.

2.22.5 JVM Stack Overflow

Platform: 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.

2.22.6 Using AWT libraries May Cause a JVM Crash

Platform: 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
```

2.23 Life Cycle Management Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.23.1, "Lifecycle Config Plugin Path May Not be Updated After Unpacking Domain"

- Section 2.23.2, "Oracle Traffic Director is Not Being Updated With Resource Group Changes"

2.23.1 Lifecycle Config Plugin Path May Not be Updated After Unpacking Domain

When packing a domain that has been configured to use Lifecycle Manager in one environment using the pack command, and unpacking it in another environment, the plugin 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.

2.23.2 Oracle Traffic Director is Not Being Updated With Resource Group Changes

In an integrated environment involving Weblogic Server, Oracle Traffic Director (OTD), and Lifecycle Manager, if the out of band functionality of Lifecycle Manager is enabled then the expectation is that any change in resource groups including targeted changes to domain partitions will be sent to OTD using Lifecycle Manager. However, as of 12.2.1, this behavior has changed; the updates to resource groups, including adding or deleting a resource group, are not being sent to OTD.

Workaround

You must restart the domain partition after targeting configuration changes. This ensures that OTD is updated with the latest configuration changes in resource groups.

2.24 Monitoring Issues and Workarounds

This section describes the following issue and workaround:

- Section 2.24.1, "MBean Attributes Not Explicitly Marked as @unharvestable Appear as Harvestable"

- Section 2.24.2, "Issue with Ambiguous Watch Rule ObjectName Patterns"

- Section 2.24.3, "Behavior Change in CreateSystemResourceControl"

- Section 2.24.4, "Errors May Occur Writing to WLDF Archive During EBR Upgrade of WLDF Schema Using Oracle Upgrade Assistant"

2.24.1 MBean Attributes Not Explicitly Marked as @unharvestable Appear as Harvestable

Platform: 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 and will appear as harvestable in the WebLogic Server Administration Console.

Workaround

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

2.24.2 Issue with Ambiguous Watch Rule ObjectName Patterns

Platform: 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.

2.24.3 Behavior Change in CreateSystemResourceControl

Platform: All

This issue is related to a change in how 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, if the Harvester and Watch elements in your deployed descriptor resemble the following:

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

and you register this descriptor with the runtime control as `createSystemControl("resource1", ...)`, previously 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 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.

2.24.4 Errors May Occur Writing to WLDF Archive During EBR Upgrade of WLDF Schema Using Oracle Upgrade Assistant

Platform: All

If you configured the WLDF archive for servers in a domain to use an Oracle EBR editioned schema, and attempt to upgrade that schema using Upgrade Assistant, it is possible that some errors may occur in servers that are still running against that schema when the upgrade is performed.

This is relevant in the following conditions:

- WLDF schemas installed to an Oracle EBR database using the 12.1.2 Repository Creation Utility (RCU) or manually created WLDF schemas from an earlier version of WebLogic Server

- When using the Upgrade Assistant, the user chooses to upgrade the existing schema to use Oracle EBR editioned tables

- There are servers running while the upgrade is performed (for example, if the schema is shared across multiple WebLogic Server domains)

During the upgrade, the WLDF tables are renamed and edition-based views are created to point to the renamed instances. No data is lost in this operation. However, in these situations, there may be a brief period where some errors can occur (during the time while the upgrade is performed) in the running servers when recording WLDF Harvester or Instrumentation data.

These errors are not critical and should not affect server performance, but may result in a lost of a small amount of monitoring data for those servers.

Workaround

To avoid the potential of this occurrence, you can halt the affected WebLogic Server domain(s) during the Oracle Database upgrade process.

2.25 Node Manager Issues and Workarounds

This section describes the following Node Manager issue and workaround:

- Section 2.25.1, "Removing Primary Interface Causes Error During Server Migration"

- Section 2.25.2, "Node Manager Not Putting Up -D64 When Starting Server Using Java Command"

- Section 2.25.3, "Oracle HTTP Server Instances Start in UNKNOWN State"

- Section 2.25.4, "Template Information For OPSS Missing In installNodeMgrSrv.cmd"

- Section 2.25.5, "New Node Manager Property Names Cannot Be Used From WLST Offline"

2.25.1 Removing Primary Interface Causes Error During Server Migration

Platform: 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 may lead to random exceptions occurring with Node Manager during server migration. If you have 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 (first one in 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 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
```

2.25.2 Node Manager Not Putting Up -D64 When Starting Server Using Java Command

Platform: HPUX IA64

There is a fundamental difference between Node Manager starting a server using a start script and Node Manager starting a server using the Java command. When using the start script, the -d64 flag is added based on some script language that detects the platform. Node Manager does not add this flag when starting a server with the Java command.

Workaround

When starting a server using Node Manager through the Java command, specify arguments such as -d64 in the `ServerStart` arguments field.

2.25.3 Oracle HTTP Server Instances Start in UNKNOWN State

Platform: All

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

2.25.4 Template Information For OPSS Missing In installNodeMgrSrv.cmd

Platform: MS Windows

The OPSS template information that is necessary to start Node Manager using `installNodeMgrSvc.cmd` is missing. As a result, `installNodeMgrSvc.cmd` cannot be used correctly in a JRF/OPSS environment. Node Manager will start without the information used to specify the correct classes to load KSS and will use JKS as the default keystore. You will notice this problem when you are setting up your environment and first run `startNodeManager.cmd`, which will load KSS.

Workaround

Edit the `installNodeMgrSvc.cmd` command before using it, copying the section from `startNodeManager.cmd`. For example:

```
set JAVA_OPTIONS=%JAVA_OPTIONS%
-Doracle.security.jps.config=C:\OracleHome11g\user_
projects\domains\mydomain\config\fmwconfig\jps-config-jse.xml
-Dcommon.components.home=C:\OracleHome1213\Oracle_Home\oracle_common
-Dopss.version=12.1.3
if NOT "%POST_CLASSPATH%"=="" (set POST_CLASSPATH=C:\OracleHome1213\Oracle_
Home\oracle_common\modules\oracle.jps_12.1.3\jps-manifest.jar;%POST_CLASSPATH%)
else (set POST_CLASSPATH=C:\OracleHome1213\Oracle_Home\oracle_
common\modules\oracle.jps_12.1.3\jps-manifest.jar)
```

2.25.5 New Node Manager Property Names Cannot Be Used From WLST Offline

Platform: 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*.

2.26 Operations, Administration, and Management Issues and Workarounds

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

2.27 Oracle Kodo Issues and Workarounds

This section describes the following Oracle Kodo issue and workaround:

- Section 2.27.1, "Value Retrieved for an Empty Byte Array Field is NULL"

2.27.1 Value Retrieved for an Empty Byte Array Field is NULL

Platform: 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.

2.28 Plug-ins Issues and Workarounds

This section describes the following issue for various WebLogic Server plug-ins:

- Section 2.28.1, "apr_socket_connection Exception Occurs When Using the IIS Plug-In"
- Section 2.28.2, "Failure to Introspect Write Protected Domains With Managed Servers"
- Section 2.28.3, "SYSPROP Enables HTTP Proxying in OVAB Studio"

2.28.1 apr_socket_connection Exception Occurs When Using the IIS Plug-In

Platform: All

Under the following circumstances, the IIS plug-in may not work, resulting in an `apr_socket_connection` error:

1. Both the IIS and 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 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.

2.28.2 Failure to Introspect Write Protected Domains With Managed Servers

Platform: 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.

2.28.3 SYSPROP Enables HTTP Proxying in OVAB Studio

Platform: All

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

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`

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 nonstandard desktop environment on Linux, you may need to set the `http_proxy` property with the `valuehost:port`.

2.29 Protocols Issues and Workarounds

There are no known Protocols issues in this release of WebLogic Server.

2.30 RMI-IIOP Issues and Workarounds

This section describes the following issue and workaround:

- Section 2.30.1, "Ant 1.7 rmic Task Incompatibility"

- Section 2.30.2, "Truncated Java Exception Stack Trace Returned to Client if EJB Invocation Fails"

2.30.1 Ant 1.7 rmic Task Incompatibility

Platform: All

Calls to the Ant version 1.7 `rmic` task automatically add a `-vcompat flag`, which is not compatible with `rmic` for Oracle WebLogic Server.

Workaround

Use either of the following workarounds if your rmic call is of the form:

```
rmic classname="com.bea.crmsimulation.legacyra.LegacyAdapter"
   base="${module_location}/core-legacy-ra/classes"
   classpath="${core.classes}" compiler="weblogic" />
```

- Add a `stubversion`

  ```
  <rmic classname="com.bea.crmsimulation.legacyra.LegacyAdapter"
     base="${module_location}/core-legacy-ra/classes"
     classpath="${core.classes}" compiler="weblogic"
     stubversion="1.2"/>
  ```

- Remove the `compiler` flag

  ```
  <rmic classname="com.bea.crmsimulation.legacyra.LegacyAdapter"
     base="${module_location}/core-legacy-ra/classes"
     classpath="${core.classes}"
  ```

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

Platform: 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
```

2.31 Security Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.31.1, "Service-side Kerberos Authentication Fails With Error 401"
- Section 2.31.2, "BAD_MAC_RECORD Error Occurs When Using JSSE-based SSL Provider"
- Section 2.31.3, "StoreBootIdentity Works Only if the Appropriate Server Security Directory Exists"
- Section 2.31.4, "Boot Time Failure Occurs With SecurityServiceException"
- Section 2.31.5, "Authentication Failure After Upgrading a Domain From WLS 6.1"
- Section 2.31.6, "InvalidParameterException Message Generated and Displayed"
- Section 2.31.7, "Enabling Both the Authentication and Passive Attributes In SAML 2.0 Service Provider Services Is an Invalid Configuration"
- Section 2.31.8, "Running the WebLogic Full Client in a Non-Forked VM"
- Section 2.31.9, "Random Number Generator May Be Slow on Machines With Inadequate Entropy"
- Section 2.31.10, "Additional Information For BEA-090402 Message"

- Section 2.31.11, "LDAP Authenticator Log Messages Show Incorrect URL"

- Section 2.31.12, "Security Errors Occur When Starting ODI Managed Server"

- Section 2.31.13, "Use of WebLogic SAML Support For SSO is Not Supported in Partitions or With Security Realm Restart"

- Section 2.31.14, "Compatibility Realm Domains Are Not Supported by WebLogic 12.2.1 Clients"

- Section 2.31.15, "Partition Creation Fails after Upgrading a Domain"

2.31.1 Service-side Kerberos Authentication Fails With Error 401

Platform: All

Service-side Kerberos authentication fails with an HTTP/1.1 `Error 401 Unauthorized` if the JDK version is JRockit 1.60_24, 1.60_28, or 1.60_29.

Workaround

Use one of the following workarounds:

- Instead of using `ktab.exe` to generate the keytab file, use another tool such as `kadmin` to generate it.

- Use `ktab.exe` to manually supply the correct kvno.

2.31.2 BAD_MAC_RECORD Error Occurs When Using JSSE-based SSL Provider

Platform: All

If WebLogic Server is configured to use the JSSE-based SSL provider, attempts to create an SSL connection may fail with a `BAD_MAC_ERROR` message.

Workaround

Install JDK 7u2 or higher and restart WebLogic Server.

2.31.3 StoreBootIdentity Works Only if the Appropriate Server Security Directory Exists

Platform: All

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

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

2.31.4 Boot Time Failure Occurs With SecurityServiceException

Platform: 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.

2.31.5 Authentication Failure After Upgrading a Domain From WLS 6.1

Platform: All

After upgrading a domain from WLS 6.1, the WebLogic Server instance will not boot due to an authentication failure.

Workaround

A system user password must be set up in the WLS 6.1 domain before or after the upgrade process in order for the WebLogic Server instance to boot properly.

2.31.6 InvalidParameterException Message Generated and Displayed

Platform: 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 and displayed in the WebLogic Server Administration Console.

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.

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

Platform: 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.

Note that the Force Authentication attribute has no effect because SAML logout is not supported in WebLogic Server. So even if the user is already authenticated at the Identity Provider site and Force Authentication is enabled, the user is not forced to authenticate again at the Identity Provider site.

Avoid enabling both these attributes.

2.31.8 Running the WebLogic Full Client in a Non-Forked VM

Platform: All

If the WebLogic Full Client is running in a non-forked VM, for example by means of a `<java>` task invoked from an Ant script without the `fork=true` attribute, the following error might be generated:

```
java.lang.SecurityException: The provider self-integrity check failed.
```

This error is caused by the self-integrity check that is automatically performed when the RSA Crypto-J library is loaded. (The Crypto-J library, `cryptoj.jar`, is in the `wlfullclient.jar` manifest classpath.)

This self-integrity check failure occurs when the client is started in a non-forked VM and it uses the Crypto-J API, either directly or indirectly, as in the following situations:

- The client invokes the Crypto-J library directly.

- The client attempts to make a T3S connection, which triggers the underlying client SSL implementation to invoke the Crypto-J API.

When the self-integrity check fails, further invocations of the Crypto-J API fail.

Workaround

When running the full client in a `<java>` task that is invoked from an Ant script, always set the `fork` attribute to `true`.

For more information about the self-integrity check, see "How a Provider Can Do Self-Integrity Checking" in How to Implement a Provider in the Java™ Cryptography Architecture, available at the following URL:

http://docs.oracle.com/javase/6/docs/technotes/guides/security/crypto/HowToImplAProvider.html#integritycheck

2.31.9 Random Number Generator May Be Slow on Machines With Inadequate Entropy

Platform: 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, providing your site can tolerate lessened security. To do this, add the `-Djava.security.egd=file:///dev/urandom` switch or `file://dev/./urandom` to the command that starts the Java process. Note that this workaround should not be used in production environments because it uses pseudo-random numbers instead of genuine random numbers.

2.31.10 Additional Information For BEA-090402 Message

Platform: All

`BEA-090402` 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` is just describing the most likely root cause, which is that the customer has modified the `boot.properties` file or the boot user password and thus 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 which 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 other, less common causes.

2.31.11 LDAP Authenticator Log Messages Show Incorrect URL

The 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 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>`

2.31.12 Security Errors Occur When Starting ODI Managed Server

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.

2.31.13 Use of WebLogic SAML Support For SSO is Not Supported in Partitions or With Security Realm Restart

Platform: All

Using SAML 1.1 and SAML 2.0 for single sign-on with web browsers and HTTP clients is not supported in partitions or for automatic realm restart.

However, the use of SAML 1.1 and SAML 2.0 with web services is fully supported with these features.

2.31.14 Compatibility Realm Domains Are Not Supported by WebLogic 12.2.1 Clients

Platform: All

In WebLogic Server version 12.2.1, support for Compatibility security, including compatibility realms, has been removed. For more information about Compatibility security and compatibility realms, see "What Is Compatibility Security?" in *Administering Security for Oracle WebLogic Server 12.1.3*.

The 6.*x* realms were deprecated in WebLogic Server version 9.0 (November 2006). WebLogic domains configured with realm adapter security providers and 6.*x* realm configuration elements are no longer supported in WebLogic Server 12.2.1. For information about the specific Compatibility security and 6.*x* realm components removed and no longer supported in WebLogic Server 12.2.1, see "Removed Functionality and Components" in *What's New in Oracle WebLogic Server 12.2.1*.

In addition, WebLogic Server 12.2.1 clients, or servers acting as a client, can no longer invoke on servers if the target domain is configured with a compatibility realm. Such an invocation throws a `ClassNotFoundException` (RealmAdapterUser) and the invocation fails.

Interoperability between WebLogic Server 12.2.1 and with prior release servers is supported if compatibility realms are not configured. See "Protocol Compatibility" in *Understanding Oracle WebLogic Server*.

2.31.15 Partition Creation Fails after Upgrading a Domain

In WebLogic Server 12.2.1, the default authorization and default role mappings have changed. The role mappings have changed to include identity domains. For example, the default role mapping for the WebLogic Sever Administrator role has changed from Group(Administrators) to AdminIDDGroup(Administrators) in order to support identity domains.

If you try to create a partition in a domain after upgrading the domain, then the partition creation fails. This is because the domain is validated to check if every security realm contains the new policies and predicates. Since, the upgraded domain contains a security realm that was created in the release prior to 12.2.1, it does not contain the new policies. Therefore, the validation fails and the partition creation fails.

Workaround

After upgrading the domain, you must perform the following tasks:

1. Create a new security realm and set the new realm as the default realm.

2. Delete the old security realm.

For information about creating a new security realm or deleting an existing security realm, see "Manage security realms" in *Administration Console Online Help*.

2.32 SNMP Issues and Workarounds

There are no known SNMP issues in this release of WebLogic Server.

2.33 Spring Framework on WebLogic Server Issues and Workarounds

This section describes the following issues and workarounds:

-   Section 2.33.1, "OpenJPA ClassFileTranformer Does Not Work When Running on JRockit"

-   Section 2.33.2, "petclinic.ear Does Not Deploy on WebLogic Server"

2.33.1 OpenJPA ClassFileTranformer Does Not Work When Running on JRockit

Platform: 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`.

2.33.2 petclinic.ear Does Not Deploy on WebLogic Server

Platform: 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.

2.34 System Component Architecture (SCA) Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.34.1, "No Support for SCA in Domain Partitions"

2.34.1 No Support for SCA in Domain Partitions

The SCA container has not been updated for use in domain partitions. Use of SCA in partitions will result in undefined behavior. If you need to run SCA applications, do not add domain partitions to your domain.

2.35 Upgrade Issues and Workarounds

This section describes the following issue:

- Section 2.35.1, "SQLIntegrityConstraintViolationException May Occur When Upgrading"
- Section 2.35.2, "SAXParseException May Occur During Reconfiguration"
- Section 2.35.3, "Manually Configure DefaultIdentityAsserter During Domain Upgrade"

2.35.1 SQLIntegrityConstraintViolationException May Occur When Upgrading

Platform: 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
```

2.35.2 SAXParseException May Occur During Reconfiguration

Platform: All

The following exception appears in the `reconfig.log` after invoking `reconfig.sh` with `log_priority=ALL`:

```
[org.xml.sax.SAXParseException; lineNumber: 3; columnNumber: 77; cvc-elt.1: Cannot
find the declaration of element 'stringSubsInfo'.]
```

2.35.3 Manually Configure DefaultIdentityAsserter During Domain Upgrade

Several WebLogic Server software features, such as Lifecycle Manager, RESTful Management Services, WLS Administration 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.

2.36 Web Applications Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.36.1, "MaxPostSizeExceededException Reported in Web Browser"

- Section 2.36.2, "Administration Console Fails to Implement session-timeout Changes"

- Section 2.36.3, "Database Connections Become Unstable When a PoolLimitSQLException Occurs"

- Section 2.36.4, "Web Page Fails to Open When Accessing It Using the SSL Port"

- Section 2.36.5, "Unable to View the Output of a JSPX Page in Internet Explorer"

- Section 2.36.6, "Unable to View the Output of SVG files in Internet Explorer 7"

- Section 2.36.7, "Deployment Plans Cannot Be Used To Override Two Descriptors"

- Section 2.36.8, "Spring Dependency Injection Not Supported on JSP Tag Handlers"

- Section 2.36.9, "503 Error When Accessing an Application With a Valid sessionid"

- Section 2.36.10, "Applications Configuring jdbc-connection-timeout-secs Fail to Deploy"

- Section 2.36.11, "HttpServletRequest getLocale and getLocales Methods Changed"

- Section 2.36.12, "WebSocket: Server Cannot Receive Messages Larger Than 4MB"

- Section 2.36.13, "Default JSP Encoding Changed to UTF-8"

- Section 2.36.14, "No Support for Annotations for JSF-based Web Applications with version lesser than 2.5"

2.36.1 MaxPostSizeExceededException Reported in Web Browser

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.

2.36.2 Administration Console Fails to Implement session-timeout Changes

Platform: All

If the `session-timeout` is configured in the `web.xml` file, any changes made to change the `session-timeout` using the WebLogic Server Administration Console do not take effect.

Workaround

Use a deployment plan to override the `session-timeout` setting.

2.36.3 Database Connections Become Unstable When a PoolLimitSQLException Occurs

Platform: All

When a `PoolLimitSQLException` 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.

2.36.4 Web Page Fails to Open When Accessing It Using the SSL Port

Platform: 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** > **Options** >**Advanced** > **Encryption tab** > **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 then 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.

2.36.5 Unable to View the Output of a JSPX Page in Internet Explorer

Platform: MS Windows

When a JSPX page is deployed and is then accessed using some versions of Internet Explorer, the XHTML source is displayed instead of the page contents. This occurs in both normal and `osjp.next` modes.

Workaround

The application users should be instructed to use Firefox or Safari to access the application.

2.36.6 Unable to View the Output of SVG files in Internet Explorer 7

Platform: MS Windows

When a page using Scalar Vector Graphics is deployed and is then accessed using Internet Explorer 7 (IE7), the source is displayed instead of the page's graphic contents. This occurs in both normal and osjp.next modes.

Workaround

Application developers should avoid using SVG graphics in their applications, as it is not natively supported in IE7. If used, a warning similar to the following should be added:

```
All current browsers, with the exception of Internet Explorer, support SVG
files. Internet Explorer requires a plug-in to display SVG files. The plug-ins
are available for free, for example, the Adobe SVG Viewer at
http://www.adobe.com/svg/viewer/install/.
```

2.36.7 Deployment Plans Cannot Be Used To Override Two Descriptors

Platform: All

Deployment plans cannot be used to override the following two descriptors during 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 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.

2.36.8 Spring Dependency Injection Not Supported on JSP Tag Handlers

Platform: All

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

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

2.36.9 503 Error When Accessing an Application With a Valid sessionid

Platform: All

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

For example, 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, they will get a 503 error.

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

Platform: All

As of 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 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.

2.36.11 HttpServletRequest getLocale and getLocales Methods Changed

Platform: All

The `HttpServletRequest` `getLocale` and `getLocales` methods changed their default behavior for getting the language tag in WebLogic Server. Before 12.1.3, the `getLocale` and `getLocales` methods return language tags according to RFC3066. Since 12.1.3, these methods return language tags according to RFC5646.

Workaround

If you want to get the language tag according to RFC3066, you need to set the `langtag-revision` element of `container-descriptor` in the `weblogic.xml` file to 3066. For example:

```
<container-descriptor>
  <langtag-revision>3066</langtag-revision>
</container-descriptor>
```

The system property `-Dweblogic.servlet.langtagRevision` can also determine the locale parsing mechanism. However, if you set a value in `langtag-revision`, that value overrides the setting in `-Dweblogic.servlet.langtagRevision`. For more information, see "langtag-revision" in *Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server*.

2.36.12 WebSocket: Server Cannot Receive Messages Larger Than 4MB

Platform: 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>
```

2.36.13 Default JSP Encoding Changed to UTF-8

Platform: All

As of WebLogic Server 12.1.3, the default value of the `encoding` element for the `jsp-descriptor` element in `weblogic.xml` is `UTF-8` for JSP pages. Prior to WebLogic Server 12.1.3, the default value for JSP encoding was `ISO-8859-1`.

Workaround

To specify `ISO-8859-1` as the encoding value for a `jsp-descriptor` element, configure the `java-charset-name` element in the `input-charset` element to `ISO-8859-1`. For more information, see "weblogic.xml Deployment Descriptor Elements" in *Developing Web Applications, Servlets, and JSPs for Oracle WebLogic Server*.

2.36.14 No Support for Annotations for JSF-based Web Applications with version lesser than 2.5

As of 12.2.1, 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 older.

Workaround

For compatibility with your Web Application that has version set to 2.5 or older, it is recommended that you upgrade the version of the Web Application deployment descriptor, `web.xml` to 3.0.

For instance your existing Web application has the version set to 2.3. You should upgrade it to 3.0 as shown in the following example:

```
<?xml version = '1.0' encoding = 'windows-1252'?>
<web-app xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd"
        version="3.0">
    ...
</web-app>
```

2.37 WebLogic Server Scripting Tool (WLST) Issues and Workarounds

This section describes the following issues and workarounds:

2.37.1 Property Names Containing '.' Characters Are Not Supported by loadProperties

Platform: 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

2.37.2 Invalid cachedir Created by Jython Causes WLST to Error Out

Platform: 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.

2.37.3 WLST Error Messages Fail to Display in European Locales

Platform: All

In European locales, when using WLST, you will see a syntax error if the error message contains a single quotation mark.

2.38 Web Server Plug-Ins Issues and Workarounds

This section describes the following issue:

- Section 2.38.1, "MOD_WLS_OHS Does Not Fail Over"

2.38.1 MOD_WLS_OHS Does Not Fail Over

Platform: 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.

2.39 Web Services and XML Issues and Workarounds

This section describes the following issues and workarounds:

- Section 2.39.1, "Sparse Arrays and Partially Transmitted Arrays Are Not Supported"

- Section 2.39.2, "WSDL Compiler Does Not Generate Serializable Data Types"

- Section 2.39.3, "Use of Custom Exception on a Callback"

- Section 2.39.4, "Cannot Use JMS Transport in an Environment That Also Uses a Proxy Server"

- Section 2.39.5, "clientgen Fails When Processing a WSDL"

- Section 2.39.6, "IllegalArgumentException When Using a Two-Dimensional XML Object in a JWS Callback"

- Section 2.39.7, "Using SoapElement[] Results in Empty Array"

- Section 2.39.8, "FileNotFound Exception When a Web Service Invokes Another Web Service"

- Section 2.39.9, "Client Side Fails to Validate the Signature on the Server Response Message"

- Section 2.39.10, "xmlcatalog Element Entity Cannot Be a Remote File or a File in an Archive"

- Section 2.39.11, "Catalog File's public Element Is Not Supported When Using XML Catalogs"

- Section 2.39.12, "Local xmlcatalog Element Does Not Work Well"

- Section 2.39.13, "External Catalog File Cannot Be Used in the xmlcatalog Element of clientgen"

- Section 2.39.14, "WS-AT Interoperation Issues With WebSphere and WebLogic Server"

- Section 2.39.15, "Occasional JAX-RS Error Message in the Server Log"

- Section 2.39.16, "WebLogic Web Services Security Signature is Not Supported For FIPS 140 Mode"

- Section 2.39.17, "Jersey Applications Cannot be Deployed Without the Servlet Init Parameter"

- Section 2.39.18, "No Support for Cloning SAF Agents for Multiple Targets When Using JAX-RPC Extension Template"

2.39.1 Sparse Arrays and Partially Transmitted Arrays Are Not Supported

Platform: All

WebLogic Server does not support Sparse Arrays and Partially Transmitted Arrays as required by the JAX-RPC 1.1 Spec.

2.39.2 WSDL Compiler Does Not Generate Serializable Data Types

Platform: 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.

2.39.3 Use of Custom Exception on a Callback

Platform: All

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

Make sure that any custom exceptions that are used in callbacks are in a package that matches the target namespace of the parent Web service.

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

Platform: All

You cannot use JMS transport in an environment that also uses a proxy server. This is because, 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.

2.39.5 clientgen Fails When Processing a WSDL

Platform: All

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

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

Platform: All

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

Workaround

Currently there is no known workaround for this issue.

2.39.7 Using SoapElement[] Results in Empty Array

Platform: 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`.

2.39.8 FileNotFound Exception When a Web Service Invokes Another Web Service

Platform: All

When Web Service A wants to invoke Web Service B, Web Service A should use the `@ServiceClient` annotation to do this. 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, 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 {
   ...
```

```
    }
```

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

Platform: 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 policy 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 two 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, `WssX509PkiPathV1Token11/>` or `<sp:WssX509PkiPathV1Token10/>` should be used, instead of `<sp:WssX509Pkcs7Token11/>` or `<sp:WssX509Pkcs7Token10/>`.

2.39.10 xmlcatalog Element Entity Cannot Be a Remote File or a File in an Archive

Platform: All

For the `xmlcatalog` element in `build.xml`, 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 instead.

2.39.11 Catalog File's public Element Is Not Supported When Using XML Catalogs

Platform: All

The `public` element in a catalog file is not supported when using the XML Catalogs feature. It is not supported to be consistent with JAX-WS EntityResolver implementation. WebLogic Server only supports defining the `system` element in a catalog file.

2.39.12 Local xmlcatalog Element Does Not Work Well

Platform: 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 out of any targets.

2.39.13 External Catalog File Cannot Be Used in the xmlcatalog Element of clientgen

Platform: 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 in-line or in an external catalog file(s), or both. In order to use an external catalog file, the `xml-commons` resolver library (`resolver.jar`) must be in your 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. In this case, however, 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.

2.39.14 WS-AT Interoperation Issues With WebSphere and WebLogic Server

Platform: 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.

2.39.15 Occasional JAX-RS Error Message in the Server Log

When shutting down a 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. ...
```

This error message is benign and can be ignored.

2.39.16 WebLogic Web Services Security Signature is Not Supported For FIPS 140 Mode

WebLogic WS-Security uses the standard RSA-SHA1 algorithm, such as `<dsig:SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1">`, for Signature Method in XML Digital Signature. This Signature Method algorithm is no longer permitted for signatures in FIPS 140-2 mode. A stronger Signature Method algorithm, such as `<dsig:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256">`, is required in FIPS 140 mode, as defined in W3C XML Digital Signature spes at `http://www.w3.org/TR/xmldsig-core1/#sec-PKCS1`. The native Web Services security does not support the extended algorithm suite that uses the RSA-SHA256 algorithm for Signature Method. As a result, WebLogic Web Services security that requires XML Digital Signature in FISP 140-2 mode is not supported.

Workaround

Oracle Web Services Manager (OWSM) supports extended algorithm suite in 12.2.1. For your applications that require XML Digital Signature for Web Services security in FIPS 140 mode, it is recommended that you switch the application to use OWSM, instead of using the native Web Services security.

2.39.17 Jersey Applications Cannot be Deployed Without the Servlet Init Parameter

The table in the Packaging the RESTful Web Service Application Using web.xml Without Application Subclass section of *Developing and Securing RESTful Web Services for Oracle WebLogic Server* states that the `<init-param>` element is not required in the servlet declaration while updating the web.xml file if the application subclass is not packaged with web.xml. The example in this section also describes the following configuration update that is required in the web.xml file if `javax.ws.rs.core.Application` is not packaged:

```
<web-app>
    <servlet>
        <servlet-name>Jersey Web Application</servlet-name>

<servlet-class>org.glassfish.jersey.servlet.ServletContainer</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>Jersey Web Application</servlet-name>
        <url-pattern>/*</url-pattern>
    </servlet-mapping>
</web-app>
```

However, this configuration fails to deploy the application as expected, though the application deployment may appear seemingly complete.

Workaround

The 12.2.1 implementation of WebLogic Server requires one of the following `<init-param>` elements to be specified in `web.xml`:

```
jersey.config.server.provider.packages
jersey.config.server.provider.classnames
jersey.config.server.provider.classpath
```

The application deployment is successful if one of the `<init-param>` elements is specified in the configuration as shown in the following example:

```
<web-app>
    <servlet>
        <servlet-name>Jersey Web Application</servlet-name>

<servlet-class>com.sun.jersey.spi.container.servlet.ServletContainer</servlet-class>
        <init-param>
          <param-name>jersey.config.server.provider.packages</param-name>
          <param-value>org.foo</param-value>
        </init-param>
    </servlet>
    <servlet-mapping>
        <servlet-name>Jersey Web Application</servlet-name>
        <url-pattern>/*</url-pattern>
    </servlet-mapping>
</web-app>
```

2.39.18 No Support for Cloning SAF Agents for Multiple Targets When Using JAX-RPC Extension Template

Platform: All

When you extend your domain that has multiple targets using the JAX-RPC extension template, the CIE processing does not support SAF agent to be cloned for multiple targets. Each instance of SAF agent is targeted to a single target.

Workaround

You must use WLST to create additional SAF agents for multiple targets. However, each instance of SAF agent can be targeted to only one target. The following example creates SAF agents for the domain, *domain1*, containing two target clusters.

```
readDomain('domain1')
cd('/')
create('ReliableWseeSAFAgent1','SAFAgent')
create('ReliableWseeSAFAgent2','SAFAgent')
assign('SAFAgent','ReliableWseeSAFAgent1','Target','SenderCluster')
assign('SAFAgent','ReliableWseeSAFAgent2','Target','ReceiverCluster')
wlsuOfflineUpdateWLSDomain()
```

2.40 WebLogic Tuxedo Connector Issues and Workarounds

This section describes the following issue and workaround:

■ Section 2.40.1, "View Classes are not Set on a Per Connection Basis"

2.40.1 View Classes are not Set on a Per Connection Basis

Platform: 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.

2.41 Documentation Errata

This section describes documentation errata:

- Section 2.41.1, "Issues With Search Function in the Samples Viewer"
- Section 2.41.2, "Japanese Text Displays in Some Search Results Topics Avitek Medical Records"
- Section 2.41.3, "HTML Pages For Downloaded Libraries Do Not Display Properly"
- Section 2.41.4, "Fixed Issue That Caused Class Version Mismatch Error in Previous Releases"

2.41.1 Issues With Search Function in the Samples Viewer

Platform: All

The **Search** function in the Samples viewer does not work when accessing the Examples documentation by selecting **Oracle WebLogic > WebLogic Server > Examples > Documentation** from the Windows **Start** menu.

Workaround

To search the Sample Applications and Code Examples, you must start the Examples server and navigate to
`http://localhost:7001/examplesWebApp/docs/core/index.html`. Click **Instructions** and then **Search**.

2.41.2 Japanese Text Displays in Some Search Results Topics Avitek Medical Records

Platform: All

The samples viewer **Search** function may sometimes return topics that display the Japanese and English versions of some Avitek Medical Records topics simultaneously.

2.41.3 HTML Pages For Downloaded Libraries Do Not Display Properly

Platform: All

After extracting the WebLogic Server documentation library ZIP files that are available from
`http://www.oracle.com/technetwork/middleware/weblogic/documentation/index.`

`html`, the HTML pages may not display properly in some cases for the following libraries:

- E12840_01 (WebLogic Server 10.3.0 documentation library)

- E12839_01 (WebLogic Server 10.3.1 documentation library)

- E14571_01 (WebLogic Server 10.3.3 documentation library)

Workarounds

For library E12840-01, after extracting the E12840_01.zip library file, if the HTML pages are not formatting correctly, perform the following steps:

1. Go to the directory in which you extracted the zip file.

2. Locate the `/global_resources` directory in the directory structure.

3. Copy the `/global_resources` directory to the root directory of the same drive.

For libraries E12839-01 and E14571-01, this issue occurs only on Windows operating systems. If the HTML pages of the extracted library are not formatting correctly, try extracting the ZIP file using another extraction option in your unzip utility. For example, if you are using 7-Zip to extract the files, select the **Full pathnames** option. Note that you cannot use the Windows decompression utility to extract the library ZIP file.

2.41.4 Fixed Issue That Caused Class Version Mismatch Error in Previous Releases

Platform: All

In previous versions of WebLogic Server, when a remote client (a call from outside the container) used IIOP and the `wlclient.jar` file to call an EJB, and that EJB returned a value object that caused a class version mismatch between the client and the EJB, the following error occurred:

```
org.omg.CORBA.BAD_PARAM: Could not find FVD class
```

This issue also occurred between WebLogic Server instances.

This issue is fixed in WebLogic Server 12.2.1.