Oracle® Application Server

Release Notes

10*g* Release 3 (10.1.3) for HP-UX PA-RISC (64-Bit)

B28068-01

February 2006

ORACLE®

Oracle Application Server Release Notes, 10*g* Release 3 (10.1.3) for HP-UX PA-RISC (64-Bit)

B28068-01

Contents

4   Oracle HTTP Server

5   Oracle Containers for J2EE

6 Oracle Application Server Portal

Preface

This preface includes the following topics:

- Documentation Accessibility
- Related Documents
- Conventions

Audience

This document is intended for users of Oracle Application Server 10*g*.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

http://www.oracle.com/accessibility/

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Related Documents

For more information, see these Oracle resources:

- Oracle Application Server Documentation on Oracle Application Server Disk 1
- Oracle Application Server Documentation Library 10*g* Release 3 (10.1.3)

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.

1

Introduction

This chapter introduces Oracle Application Server Release Notes, 10*g* Release 3 (10.1.3). It includes the following topics:

- Section 1.1, "Latest Release Information"
- Section 1.2, "Purpose of this Document"
- Section 1.3, "Operating System Requirements"
- 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/technology/documentation/

1.2 Purpose of this Document

This document contains the release information for Oracle Application Server 10*g* Release 3 (10.1.3). It describes differences between Oracle Application Server 10*g* Release 3 (10.1.3) and its documented functionality.

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

1.3 Operating System Requirements

Oracle Application Server installation and configuration will not complete successfully unless users meet the hardware and software pre-requisite requirements before installation. Refer to *Oracle Application Server Installation Guide* for a complete list of operating system requirements.

1.4 Certification Information

The latest certification information for Oracle Application Server 10*g* Release 3 (10.1.3) is available at:

http://metalink.oracle.com

1.5 Licensing Information

Licensing information for Oracle Application Server 10*g* Release 3 (10.1.3) is available at:

http://oraclestore.oracle.com

Detailed information regarding license compliance for Oracle Application Server 10*g* Release 3 (10.1.3) is available at:

http://www.oracle.com/technology/products/ias/index.html

2

Installation and Upgrade Issues

This chapter describes installation and upgrade issues and their workarounds associated with Oracle Application Server. It includes the following topics:

- Section 2.1, "Upgrade Issues"

2.1 Upgrade Issues

This section describes issues with upgrade of Oracle Application Server. It includes the following topic:

- Section 2.1.1, "Additional Data Source Requirement for OEMS JMS Database Applications"

2.1.1 Additional Data Source Requirement for OEMS JMS Database Applications

If you are deploying an OEMS JMS Database application on Oracle Application Server 10g Release (10.1.3), note that you must verify that the manage-local-transactions attribute in the data-sources.xml file is set to false.

The following example shows the managed-data-source element in the data-sources.xml file with the required attribute for OEMS JMS Database applications:

```
<managed-data-source name="OracleDS" connection-pool-name="Example
Connection Pool" jndi-name="jdbc/OracleDS" *manage-local-transactions="false"*/>.
```

3

General Management and Security Issues

This chapter describes management and security issues associated with Oracle Application Server. It includes the following topics:

- Section 3.1, "General Issues and Workarounds"
- Section 3.2, "Clustering and Replication Issues"

3.1 General Issues and Workarounds

This section describes general management and security issues. It includes the following topics:

- Section 3.1.1, "Limited Management Support for Multiple-JVM OC4J Instances"
- Section 3.1.2, "OC4J Restart Required When Changing the Name or URL of a JDBC Data Source or Connection Pool"
- Section 3.1.3, "Problem Removing a Property from a Native Data Source"
- Section 3.1.4, "Important Restriction When Setting Thread Pool Size on the Thread Pool Configuration Page"
- Section 3.1.5, "Use the Cluster Topology Page to Restart the OC4J Instance"
- Section 3.1.6, "TopLink Sessions Not Available in Application Server Control Console"
- Section 3.1.7, "Unable to Receive MBean Notification Using OPMN to Start or Stop OC4J"
- Section 3.1.8, "Using the Java Server Pages Standard Tag Libraries"
- Section 3.1.9, "Error While Generating Web Service"
- Section 3.1.10, "Problem with Deployment of non-English Character Java Server Pages"
- Section 3.1.11, "RMD Conditional Does Not Fully Evaluate"

3.1.1 Limited Management Support for Multiple-JVM OC4J Instances

With Oracle Application Server 10*g* Release 3 (10.1.3), you can configure OC4J to use multiple Java Virtual Machines (JVMs) by setting the `numprocs` argument in the `opmn.xml` file to a number greater than one (1).

For example:

```
<ias-component id="OC4J">
    <process-type id="home" module-id="OC4J" status="enabled">
```

```
                              .
                              .
                              .
                <process-set id="default_group" numprocs="2"/>
            </process-type>
    </ias-component>
```

However, this feature is not supported by Oracle Enterprise Manager 10*g* Application Server Control (Application Server Control). Instead, when the numprocs argument is set to more than one (1), you must use command line tools to manage your Oracle Application Server environment. For example, you must use:

- admin_client.jar for deployment, re-deployment, undeployment, start and stop applications, and shared library management

- Apache Ant for deployment, redeployment, and undeployment of your applications

- opmnctl commands for starting, stopping, and other life cycle operations on the Oracle Application Server

For all other administrative configuration changes, if you are using multiple JVMs, you must shut down Oracle Application Server, manually configure the relevant XML files, and then restart Oracle Application Server.

3.1.2 OC4J Restart Required When Changing the Name or URL of a JDBC Data Source or Connection Pool

If you modify the name or the connection URL of a JDBC data source or JDBC connection pool, then you must restart the OC4J instance; otherwise the changes you make will not take effect.

For example, if you use the JDBC Resources page in the Application Server Control Console to change the connection URL of a JDBC connection pool, you will not be prompted to restart the OC4J instance, but the restart is required. If you do not restart the OC4J instance, any deployed applications that require the data source will attempt to use the original connection URL.

> **See Also:** "Managing Data Sources and JDBC Connection Pools" in the Application Server Control online help

3.1.3 Problem Removing a Property from a Native Data Source

If you use the Application Server Control Console to remove a property from a native data source, Enterprise Manager does not remove the property from the underlying connection factory. As a result, the property (and its current value) is not changed.

This is expected behavior. To set a value on the underlying connection factory, use the setProperty operation of the JDBCDataSource MBean for the native Data Source to do this. You can use the MBean Browser, which is available in the Application Server Control Console, to invoke an MBean operation.

> **See Also:** "About the MBean Browser" in the Application Server Control online help

3.1.4 Important Restriction When Setting Thread Pool Size on the Thread Pool Configuration Page

By default, the Thread Pool Configuration page in the Application Server Control Console shows a minimum thread pool size of one (1) for an unconfigured thread pool. This value is expected, but do not click **OK** on this screen with a minimum thread pool size set to one (1) for any thread pool.

If you are configuring OC4J thread pools using the Thread Pool Configuration Administration Task, be sure to replace the default value for the minimum thread pool size to a number that is greater than 10; otherwise, you will not be able to login to the Application Server Control Console the next time you restart the OC4J instance.

In the event that you set the minimum thread pool size to 1 and then click **OK** on the Thread Pool Configuration page, and as a result, you cannot access the Application Server Control Console, do the following:

1. Stop the OC4J instance using the command line.

    For example, on a UNIX system, in a managed, Oracle Application Server environment, enter the following command:

    ```
    ORACLE_HOME/opmn/bin/opmnctl stopall
    ORACLE_HOME\opmn\bin\opmnctl stopall
    ```

    > **See Also:** "Starting and Stopping" in the *Oracle Application Server Administrator's Guide*
    >
    > "Starting and Stopping OC4J" in the *Oracle Containers for J2EE Configuration and Administration Guide*

2. Edit the `server.xml` file so that all thread pools are configured with a minimum pool size that is greater than 10.

    The server.xml for the OC4J instance is located in the following directory:

    ```
    ORACLE_HOME/OC4J_instance_name/config/
    ```

    For example:

    ```
    ORACLE_HOME/home/config/
    ```

Alternatively, to reset thread pool configuration to the default to factory settings, remove the `<global-thread-pool>` element and the `<work-manager-thread-pool>` element from the `server.xml` file.

3.1.5 Use the Cluster Topology Page to Restart the OC4J Instance

Some OC4J configuration pages in the Application Server Control Console (including the JTA Administration and Oracle Internet Directory Association pages) require a restart of the OC4J instance for changes to take affect.

If you use the **Restart** link, which is displayed after applying changes to one of these pages, the operation may take a few minutes because it performs an internal restart of the OC4J instance. Instead of using the **Restart** link, Oracle recommends that users navigate to the Cluster Topology page, select the affected OC4J instance, and then click **Restart** to perform a full restart of the OC4J instance. In a standalone OC4J environment, Oracle recommends that users use the command line to restart the OC4J instance.

3.1.6 TopLink Sessions Not Available in Application Server Control Console

If the TopLink Sessions for a TopLink-enabled application are not available in Application Server Control Console, check to be sure the TopLink session is configured to create the MBeans at login time. This is done by ensuring that the application has a `serverPlatform` class defined, and that the `ServerPlatform` class has its is `RuntimeServicesEnabled` flag enabled.

For Oracle Application Server 10*g* Release 3 (10.1.3), you should be using the following platform class, which can be set in the `sessions.xml` or through the session API:

```
oracle.toplink.platform.server.oc4j.Oc4j_10_1_3_Platform
```

When developing a TopLink-enabled application using Oracle JDeveloper, make sure to use version 11 or higher.

> **See Also:** "Configuring the Server Platform" in the *Oracle TopLink Developer's Guide*

3.1.7 Unable to Receive MBean Notification Using OPMN to Start or Stop OC4J

You will not be able to receive notification from the `ias:j2eeType=J2EEServer,name...` MBean entity if you start or stop Oracle Containers for J2EE (OC4J) using OPMN. This happens using either the Application Server Control or the `opmnctl stop` or `opmnct start` command from the command line.

There is presently no workaround for this issue.

3.1.8 Using the Java Server Pages Standard Tag Libraries

The Java Server Pages Standard Tag Library (JSTL) makes use of Jaxp 1.2 classes that are packaged with Java Developer Kit 1.4.

Oracle Application Server 10*g* Release 3 (10.1.3) makes use of JDK 1.5 which uses Jaxp 1.3 classes. However, the JSTL still requires the Jaxp1.2 classes. If you run the JSTL with XML related tags in JDK 1.5 you may receive an error message similar to:

```
: missing class org.apache.xpath.encounter failure.
```

To avoid JSTL failure, include the `xalan.jar` file in the required `.war` file. Add the `xalan.jar` file into your `/WEB-INF/lib` directory with the `.war` file and then re-package.

For more information refer to the JSTL release notes at:

http://java.sun.com/webservices/docs/1.6/jstl/ReleaseNotes.html.

3.1.9 Error While Generating Web Service

When generating a Web Service from a stored procedure including XSL transformation, and returned data contain non-English characters, an error will be thrown as follows:

```
java.sql.SQLException: Invalid UTF8 encoding.
```

To workaround around this issue, in the `oracle/j2ee/ws/tools/wsa/db/webservices10literal.properties` file of *ORACLE_HOME*`/webservices/lib/wsa.jar` home, modify:

```
org.w3c.dom.Document _tmpDocument_ =db.parse(%1.getclobval().getAsciiStream());
```

to:

```
db.parse(new org.xml.sax.InputSource(__jRt_0.getclobval().getCharacterStream()));
```

3.1.10 Problem with Deployment of non-English Character Java Server Pages

When you deploy an application with non-English page character set Java Server Pages (JSPs) using a Web browser, accessing the JSPs leads to corresponding modification to the encoding setting of the Java-compiler. However, if you deploy applications which use the Java-compiler to compile auto-generated codes containing non-English characters (for example, an CMP application containing non-English characters as column names), an exception will be thrown as follows:

```
@ com.evermind.compiler.CompilationException: Syntax error in source or
compilation failed.
```

To workaround this problem:

1. Shutdown OC4J

2. Remove `encoding="<SOME-ENCODING>"` from the `<java-compiler>` element in the *ORACLE_HOME*`/j2ee/home/comfig/server.xml` file.

3. Restart OC4J.

3.1.11 RMD Conditional Does Not Fully Evaluate

As documented in the *Oracle Process Manager and Notification Server Administrator's Guide* and functional specifications for Dynamic Resource Management (DRM), a Resource Management Directive (RMD) conditional can have a fully qualified path. However, the conditional may not evaluate at all. It may fail to trigger any action or exception even though the opmn.xml file is valid.

RMD definitions can be either:

- Hierarchical: if defined at the `ias-instance` level or lower. Hierarchical RMDs assume an association within the OPMN configuration components in which they are defined.

- Global: if defined at the `process-manager` level. Global RMDs require explicit OPMN component specifications.

If you are referencing a hierarchical RMD, instead of a fully qualified path use a hierarchical relative reference.

For example, if the average request time is greater than 500 milliseconds for at least 60 seconds and there are less then 4 processes running for the `process-set` at which the hierarchical RMD was configured for OC4J, you would use the following in the `opmn.xml` file:

```
([process].avgReqTime > 500 {duration(60)})&([process-set].numProcs < 4)
```

If you are referencing a global RMD use a global absolute reference.

For example, if the heap size of a Java Virtual Machine (JVM) has exceeded 500 MBs, you would use the following in the `opmn.xml` file:

```
[process-set=home][process].heapSize > 500000
```

3.2 Clustering and Replication Issues

This section describes clustering and replication issues. It includes the following topics:

- Section 3.2.1, "State Replication Framework"
- Section 3.2.2, "Using Oracle Universal Installer Provided Sample Cluster Discovery Address May Inadvertently Cluster Servers"
- Section 3.2.3, "Configuration of Oracle Application Server Clusters"

3.2.1 State Replication Framework

The state replication framework that is used by Oracle Application Server has been upgraded with several fixes that resolve multicast state replication issues and enable redeployment of running applications that use the framework.

Before deploying applications using the state replication framework in a production environment, go to Oracle*MetaLink* (`http://metalink.oracle.com`) and download and apply the required patch for Bug 4685049.

3.2.2 Using Oracle Universal Installer Provided Sample Cluster Discovery Address May Inadvertently Cluster Servers

Oracle Universal Installer provides an example cluster discovery address as part of the advanced installation option. The provide example discovery address is `225.0.0.1:6789`. This *is not* a recommended address; rather it is an example intended to provide the type of cluster discovery address users may ask for from their network administrator.

Because the cluster configuration of Oracle Application Server is fully dynamic it is possible for installations using the example cluster discovery address (`225.0.0.1:6789`) to be inadvertently clustered with other servers installed with the same example cluster discovery address.

The cluster discovery address of a specific Oracle Application Server instance can be set from the command line using the following `opmnctl` command:

```
> $ORACLE_HOME/opmn/bin/opmnctl config topology update discover=<cluster config
address>
```

For example, to update a cluster discovery address in a specific Oracle Application Server instance to be `225.0.0.1:9876`, the command would be:

```
> $ORACLE_HOME/opmn/bin/opmnctl config topology update discover="*225.0.0.1:9876"
```

Details on configuring topologies and the cluster discovery address can be found in Chapter 8, "Configuring and Managing Clusters" of the *Oracle Containers for J2EE Configuration and Administration Guide*.

3.2.3 Configuration of Oracle Application Server Clusters

Oracle Application Server instances can be grouped together in clusters using the *ORACLE_HOME*/bin/opmnassociate command line utility or explicitly using the more comprehensive *ORACLE_HOME*/opmnctl command line tool.

Additional information on post-installation topology and cluster configuration can be found in Chapter 8, "Configuring and Managing Clusters"" of the *Oracle Containers for J2EE Configuration and Administration Guide*.

4

Oracle HTTP Server

This chapter describes issues associated with Oracle HTTP Server. It includes the following topics:

- Section 4.1, "Documentation Errata"

4.1 Documentation Errata

This section describes documentation errata. It includes the following topics:

- Section 4.1.1, "Default Values for Oc4jCacheSize"
- Section 4.1.2, "UseOutputStreamSize"

4.1.1 Default Values for Oc4jCacheSize

The "Understanding Modules" chapter of the *Oracle HTTP Server Administrator's Guide* contains the default value of 1 for Oc4jCacheSize.

The default value for `Oc4jCacheSize` should be 1.

4.1.2 UseOutputStreamSize

The "Using Oracle Containers for J2EE" appendix of the *Oracle HTTP Server Administrator's Guide* has a "Configuring OC4J Plug-in on Sun ONE" section that has the following example:

```
Service type="oracle/opii" fn="opii_service" UseOutStreamSize=8192
```

It should be:

```
Service type="oracle/opii" fn="opii_service" UseOutputStreamSize=8192
```

5

Oracle Containers for J2EE

This chapter discusses release notes for Oracle Containers for J2EE (OC4J) for 10.1.3. It includes the following topics:

- Section 5.1, "Configuration, Deployment, and Administration"
- Section 5.2, "Release Notes for Servlets"
- Section 5.3, "Release Notes for EJB"
- Section 5.4, "Release Notes for Web Services"
- Section 5.5, "Release Notes for Web Services Security"
- Section 5.6, "Release Notes for OC4J Services"
- Section 5.7, "Release Notes for J2EE Connector Architecture (J2CA)"
- Section 5.8, "Release Notes for OracleAS JAAS Provider and Security"
- Section 5.9, "Release Notes for Documentation Errata"
- Section 5.10, "Oracle Application Server Containers for J2EE Job Scheduler"

You can access Oracle manuals mentioned in this document at the following URL:

http://www.oracle.com/technology/index.html

5.1 Configuration, Deployment, and Administration

This section describes configuration, deployment, and administration issues for Oracle Application Server Containers for J2EE (OC4J). This section covers the following topic(s):

- Section 5.1.1, "JDK 5.0 Requires the -doCloseWithReadPending Option"
- Section 5.1.2, "Deprecated Environment Variables dedicated.connection, dedicated.rmicontext, and LoadBalanceOnLookup"
- Section 5.1.3, "Deprecated Environment Variable ejb.batch.compile"
- Section 5.1.4, "Deprecated orion-ejb-jar.xml Attributes"
- Section 5.1.5, "Web-Site-Related Options No Longer Available"
- Section 5.1.6, "Unsupported Methods in JMX MBeanServer and MBeanServerConnection Interfaces"
- Section 5.1.7, "Upgrade to Latest J2SE Release"
- Section 5.1.8, "Workaround for ORA-604/ORA-12705 Error Using a Not-Fully Supported Locale"

- Section 5.1.9, "Incompatibility When Moving Between JDK 1.5 and 1.4"

- Section 5.1.10, "Configuring a Machine to Work With and Without a Network Connection"

- Section 5.1.11, "Converting Pre-10.1.3 Data Sources to 10.1.3 Format"

- Section 5.1.12, "Xalan Library Not Supported as a Shared Library with JDK1.4"

- Section 5.1.13, "Recommendation for <cluster> Element write-quota Setting"

- Section 5.1.14, "Configuring OC4J OCI Drivers"

For information on configuring OC4J, see the Configuration Guide for OC4J at:

http://www.oracle.com/technology/index.html

5.1.1 JDK 5.0 Requires the -doCloseWithReadPending Option

The -doCloseWithReadPending option in Java and JRE versions 1.5 or 5.0 allows one thread to close a socket when there is a read pending on the same socket from another thread.

When close() is called on a socket which has an outstanding read call from another

thread, the close() by default blocks the socket until the read call completes.

With the -doCloseWithReadPending option, the socket close() call closes the socket and in the context of the thread with the pending read, a SocketException with the message "Socket closed" is thrown.

5.1.2 Deprecated Environment Variables dedicated.connection, dedicated.rmicontext, and LoadBalanceOnLookup

Environment variables dedicated.connection, dedicated.rmicontext, and LoadBalanceOnLookup are deprecated.

To configure replication-based load balancing, use environment variable oracle.j2ee.rmi.loadBalance with the settings that Table 5–1 lists.

Table 5–1 Settings for Environment Variable oracle.j2ee.rmi.loadBalance

Setting	Description
client	The client interacts with the OC4J process that was initially chosen at the first lookup for the entire conversation (Default)
context	The client goes to a new server when a separate context is used (similar to deprecated dedicated.rmicontext).
lookup	The client goes to a new server for every lookup.

5.1.3 Deprecated Environment Variable ejb.batch.compile

Environment variable ejb.batch.compile is deprecated.

To enable or disable batch compilation, use the orion-application.xml file <orion-application> element batch-compile attribute.

5.1.4 Deprecated orion-ejb-jar.xml Attributes

The following `orion-ejb-jar.xml` file attributes are deprecated:

- `max-instances-per-pk`

- `min-instances-per-pk`

- `disable-wrapper-cache`

- `instance-cache-timeout`

- `locking-mode="old_pessimistic"`

> **Note:** Do not use these attributes in this release. Doing so will lead to deployment failure.

5.1.5 Web-Site-Related Options No Longer Available

The OC4J web-site-related options (accessible with the `-site` command) that were provided in the `admin.jar` utility in previous releases are no longer available.

For information on how to create and manage OC4J web-site configurations see the"Managing Web Sites in OC4J" chapter in the *Oracle Containers for J2EE Configuration and Administration Guide* .

5.1.6 Unsupported Methods in JMX MBeanServer and MBeanServerConnection Interfaces

A number of methods from the JMX `MBeanServer` interface are not available to J2EE applications when they use the `MBeanServer` object obtained from the following operation:

```
MBeanServer mbsrv = MBeanServerFactory.newMBeanServer();
```

The use of any of the following methods on the returned `MBeanServer` object will throw an `UnsupportedOperationException` exception:

```
public final ClassLoader getClassLoaderFor(ObjectName mbeanName)

public final ClassLoader getClassLoader(ObjectName loaderName)

public final ClassLoaderRepository getClassLoaderRepository()

public final Object instantiate(String className)

public final Object instantiate(String className, ObjectName loaderName)

public final Object instantiate(String className, Object[] params, String[]
signature)

public final Object instantiate(String className, ObjectName loaderName, Object[]
params, String[] signature)

public final ObjectInstance createMBean(String className, ObjectName name)

public final ObjectInstance createMBean(String className, ObjectName name,
ObjectName loaderName)
```

```
public final ObjectInstance createMBean(String className, ObjectName name,
Object[] params, String[] signature)

public final ObjectInstance createMBean(String className, ObjectName name,
ObjectName loader, Object[] params, String[] signature)

public final ObjectInputStream deserialize(ObjectName name, byte[] )

public final ObjectInputStream deserialize(String className, byte[] data)

public final ObjectInputStream deserialize(String className, ObjectName
loaderName, byte[] data)
```

A number of methods from the `MBeanServerConnection` interface are not supported when an application uses the Oracle JMX connectors. The use of any of the following methods on the `MBeanServerConnection` object that is created will throw an `UnsupportedOperationException` exception:

```
public final ObjectInstance createMBean(String className, ObjectName name)

public final ObjectInstance createMBean(String className, ObjectName name,
ObjectName loaderName)

public final ObjectInstance createMBean(String className, ObjectName name,
Object[] params, String[] signature)

public final ObjectInstance createMBean(String className, ObjectName name,
ObjectName loader, Object[] params, String[] signature)
```

5.1.7 Upgrade to Latest J2SE Release

Currently Oracle Application Server 10.1.3.0.0 is certified with JDK 1.4.2_08 and JDK 1.5.0_02. The product installs with JDK 1.5.0_02 by default.

In general, J2SE releases are number *a.b.c_d*, where "*a.b.c*" is the major release number, as in 1.4.2 or 1.5.0, and "*d*" is the minor release number, as in "05" or "06". As a general practice, Oracle recommends that customers upgrade to the latest minor release number of J2SE to ensure that they benefit from any bugs resolved in those specific J2SE upgrades. Oracle explicitly restates the certification matrix for major release numbers of J2SE.

Currently there is a known J2SE bug in J2SE 1.5.0_05 and J2SE 1.5.0_06 that manifests itself in an out-of-memory error in long-running stress tests involving `BigDecimals` numeric types. This bug is tracked by Sun at:

http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=6360541
http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=6372116

The workaround for this bug is to upgrade to J2SE 1.5.0_06 and set the JVM startup parameters for the impacted Oracle Containers for J2EE instance with this additional parameter:

```
-XX:CompileCommand=exclude,oracle/jdbc/driver/NumberCommonaccessor.getBigdecim
```

Information on configuring the J2SE runtime in Oracle Application Server can be found in the *Oracle Containers for J2EE Configuration and Administration Guide* at:

http://www.oracle.com/technology/index.html

5.1.8 Workaround for ORA-604/ORA-12705 Error Using a Not-Fully Supported Locale

When you try to get a connection on a Locale that is not supported in JDBC, JDBC throws a SQLException. - Bug 4704421

Use the following to verify runtime Java's locale:

```
System.out.println(Locale.getDefault().toString())
```

Unsupported Locales include any Locale that is NOT listed in the "Fully Supported Locales" table in on the Java 5.0 Java Supported Locales page at the following URL: http://java.sun.com/j2se/1.5.0/docs/guide/intl/locale.doc.html

For example, Locales that are "Provided but not Tested" include the following:

- ab_CD
- fr_FR_EURO
- it_IT_EURO
- th_TH_TH, Thai (Thailand,TH)
- be, Belarusian
- be_BY, Belarusian (Belarus)
- es_AR, Spanish (Argentina)
- es_BO, Spanish (Bolivia)
- es_DO, Spanish (Dominican Republic)
- es_EC, Spanish (Ecuador)
- es_HN, Spanish (Honduras)
- es_PY, Spanish (Paraguay)
- es_UY, Spanish (Uruguay)
- mk, Macedonian
- mk_MK, Macedonian (Macedonia)
- no_NO_NY, Norwegian (Norway,Nynorsk)
- sq, Albanian
- sq_AL, Albanian (Albania)

The workaround for this problem is to update your default Locale setting of Java. You can do any of the following:

- Change default Locale from the unsupported `es_AR`, to the fully supported `es` (Spanish):

  ```
   Locale.setDefault(new Locale("es"));
  ```

- When Locale has variant code such as `fr_FR_EURO`, remove variant code (EURO) and set default:

  ```
  Locale.setDefault(new Locale("fr", "FR"));
  ```

- Set English as the default Locale:

  ```
  Locale.setDefault(Locale.ENGLISH);
  ```

5.1.9 Incompatibility When Moving Between JDK 1.5 and 1.4

When you deploy an application (including the OC4J default application) to OC4J running JDK 1.5 (Java 5), you cannot re-use that deployment on OC4J running JDK 1.4.

Code compiled with JDK 1.5 (Java 5) cannot be read by the JDK 1.4 VM. When OC4J is running under JDK 1.4 and tries to load a class which was compiled with JDK 1.5, a class loading exception will be thrown with the following message:

```
Unsupported major.minor version 49.0
```

This can occur in scenarios such as:

- You deploy an application that contains EJBs to OC4J running under JDK 1.5, then, without undeploying the application, you restart OC4J under JDK 1.4. The problem is that the generated code associated with the EJBs will have been compiled with the same JDK version that was used to start the server and that the generated code is cached between server restarts on the file system in the OC4J_HOME/j2ee/home/application-deployments directory.

  The workaround for this is to shutdown the server, remove either the contents of the *OC4J_HOME*/j2ee/home/application-deployments directory (or just the offending application's sub-directory) and restart the server with JDK 1.4.

- You deploy an EAR file which contains classes that were compiled with and targeted to JDK 1.5 to OC4J running under JDK 1.4.

  The workaround for this is to recompile the contents of the EAR using JDK 1.4 and redeploy.

> **Note:** To simplify this discussion, we assume that no cross compilations are being used to target code to specific JDK versions.

5.1.10 Configuring a Machine to Work With and Without a Network Connection

When you work on a single machine using localhost, add the IP address in the `<ipaddr>` subelement of the `<notification-server>` element and explicitly set up a discover list in the `<discover>` element to refer to the localhost OPMN remote port, as defined in the cluster `<port>` element. An example of this configuration follows:

```
<notification-server>
    <ipaddr remote="127.0.0.1" request="127.0.0.1"/>
    <port local="6101" remote="6201" request="6004"/>
    <ssl enabled="true"
wallet-file="$ORACLE_HOME\opmn\conf\ssl.wlt\default"/>
    <topology>
        <discover list="localhost:6201"/>
    </topology>
</notification-server>
```

If you supply the localhost IP address, 127.0.0.1, the machine can work with or without a network.

5.1.11 Converting Pre-10.1.3 Data Sources to 10.1.3 Format

For information on converting pre-10.1.3 data sources to 10.1.3 format, see the release note at Section 5.6.3.3, "Converting Existing Data Sources to Release 3 Format".

5.1.12 Xalan Library Not Supported as a Shared Library with JDK1.4

If you are using JDK1.4, Oracle Application Server 10.1.3 does not support using the Xalan library shipped with the JDK as a shared library. To use the Xalan library, you have two alternatives:

- Use JDK1.5, in which the embedded Xalan library IS supported as a shared library.
- With JDK1.4, use a standalone distribution of the Xalan library instead of the embedded version.

5.1.13 Recommendation for <cluster> Element write-quota Setting

Chapter 9 of the 10.1.3 *Oracle Containers for J2EE Configuration and Administration Guide* documents the `<cluster>` element of the `orion-application.xml` file, including use of the `write-quota` attribute. This attribute determines the number of other "group members" within a cluster to which application state information should be replicated.

Be aware, however, that a "group member" is actually a JVM, not a node, and that it is possible to have multiple JVMs per node.

To ensure that more than one node receives state replication, set `write-quota` to a number greater than the highest number of JVMs on any one node within the cluster. For example, if there are three nodes, which have six JVMs, four JVMs, and three JVMS, respectively, set `write-quota` to a value of at least `7`.

5.1.14 Configuring OC4J OCI Drivers

If any application uses OCI drivers for a datasource connection, then add the following in `opmn.xml` for that particular OC4J instance. For example, if the application is deployed to the default home instance, configure the OCI drvier as follows:

```
<ias-component id="OC4J">
<environment>
<variable id="SHLIB_PATH" value="$ORACLE_HOME/lib32" />
</environment>
<process-type id="home" module-id="OC4J" status="enabled">
```

5.2 Release Notes for Servlets

This section describes release notes for servlets. It covers the following topic(s):

- Section 5.2.1, "Servlet Invocation by Classname Disabled by Default"
- Section 5.2.2, "Physical File Required for Welcome File"
- Section 5.2.3, "Warning Issued for servlet.init() Not Working with run-as"
- Section 5.2.4, "Request Parameters Not Available During Filter Execution"

5.2.1 Servlet Invocation by Classname Disabled by Default

In the 10.1.3 implementation, servlet invocation by class name is not enabled by default. Therefore, in default mode, you must use standard servlet configuration in `web.xml` before a servlet can be invoked. For example:

```
<servlet>
    <servlet-name>mytest</servlet-name>
    <servlet-class>mypackage.MyTestClass</servlet-class>
</servlet>
...
<servlet-mapping>
    <servlet-name>mytest</servlet-name>
    <url-pattern>/servlet/mytest</url-pattern>
</servlet-mapping>
```

Without this configuration, attempts to invoke the servlet will result in a `404 NOT FOUND` error. This differs from the default behavior in previous releases, where invocation by class name was enabled.

Alternatively, customers can choose to enable invocation by class name when they start OC4J, by setting the `http.webdir.enable` property as follows:

`-Dhttp.webdir.enable=true`

5.2.2 Physical File Required for Welcome File

A physical file must be present for a welcome file to dispatch to a servlet. To create a servlet mapped to `/index.html` that maps to the JSP `/index.jsp` and have it serve as a welcome file, the `web.xml` file should include the following entries:

```
<servlet>
    <servlet-name> index_jsp </servlet-name>
    <jsp-file> /index.jsp </jsp-file>
</servlet>

<servlet-mapping>
    <servlet-name>index_jsp</servlet-name>
    <url-pattern>/index.html</url-pattern>
</servlet-mapping>
```

This works *only* if there is a physical file, `/index.html`, in the Web application. The file can be zero length. As long as the file exists, this servlet will be loaded as the welcome file. Otherwise, a `java.lang.StringIndexOutOfBoundsException` exception will be thrown.

5.2.3 Warning Issued for servlet.init() Not Working with run-as

For a Web application, when `run-as user` is specified in the `web.xml` file, all method invocations except the `Servlet.init()` method will be invoked as the specified user. With the JMS Router being the default application of OC4J, calls need to be authorized to the router's EJBs. This is done by defining the application role `"jmsRouter"`, which is mapped to the JAAS `"oc4j-administrators"` role, and specifying `<method-permission>` for all methods of the router's EJBs.

The `init()` method of the servlet within the router's Web model creates a router EJB object. Regardless of whether `run-as` is specified in `web.xml` for the servlet, a security exception is thrown:

```
@ oracle.oc4j.rmi.OracleRemoteException: anonymous is not allowed to call this EJB
method, check your security settings (method-permission in ejb-jar.xml and
security-role-mapping in orion-application.xml).
```

Workaround

The security warning can be removed by commenting out `'*'` in the
`<method-name>` element of `<method-permission>` in `ejb-jar.xml` and
explicitly enumerating all methods in `AdminMgrBean` that the `jmsRouter` role can
access, as follows.

```
<!--
    <method-permission>
      <role-name>jmsRouter</role-name>
      <method>
          <ejb-name>AdminMgrBean</ejb-name>
          <method-name>*</method-name>
      </method>
    </method-permission>
-->
    <method-permission>
      <role-name>jmsRouter</role-name>
      <method>
          <ejb-name>AdminMgrBean</ejb-name>
          <method-name>getConfig</method-name>
      </method>
    </method-permission>
    ...
```

`runAsRoleName` is correctly parsed in `ServletDescriptor.java`, stored in `info`
and `thread` in `HttpApplication.loadServlet()`.

5.2.4 Request Parameters Not Available During Filter Execution

HTTP request parameters will not be available to servlet filters that are meant to be
executed before dispatch of the request to a static resource (an `.html` file, for
example). Note that filters that execute before dynamic resources, such as a servlet or
JSP, will have access to the parameters.

5.3 Release Notes for EJB

This section describes release notes for EJB. It covers the following topics:

- Section 5.3.1, "EJB 3.0 Support"

- Section 5.3.2, "Orion CMP is Deprecated"

- Section 5.3.3, "Orion CMP and Non-Oracle Databases"

- Section 5.3.4, "Stateful Session Bean Replication Trigger Configuration"

- Section 5.3.5, "EJB 3.0 Entities and Application Server Control"

- Section 5.3.6, "Entity and Session Deployment Attribute tx-retry-wait"

5.3.1 EJB 3.0 Support

In this release, OC4J supports a subset of the functionality specified in the EJB 3.0
proposed final draft at
http://jcp.org/aboutJava/communityprocess/pr/jsr220/index.html

For example, support for some EJB 3.0 features such as persistence API, external lifecycle listener class, and interceptors may not be compliant with the latest EJB 3.0 specification.

You may need to make code changes to your EJB 3.0 OC4J application after the EJB 3.0 specification is finalized and OC4J is updated to full EJB 3.0 compliance.

5.3.2 Orion CMP is Deprecated

The Orion persistence manager is deprecated. Oracle recommends that you use OC4J and the TopLink persistence manager for new development. Using the migration tool, you can easily migrate an existing OC4J application that uses EJB 2.0 entity beans with the Orion persistence manager to use EJB 2.0 entity beans with the TopLink persistence manager.

For more information, see "Migrating OC4J Orion Persistence to OC4J TopLink Persistence" in the *Oracle TopLink Developer's Guide*.

5.3.3 Orion CMP and Non-Oracle Databases

When using the (deprecated) Orion persistence manager with CMP and a non-Oracle database, OC4J does not read the schema XML file specified by the `data-sources.xml` file `managed-data-source` element `schema` attribute.

For example, consider the `data-sources.xml` and `orion-ejb-jar.xml` files shown in the following examples:

Example 5–1   Non-Oracle Database data-sources.xml

```
<connection-pool
    name="ConnectionDB2"
    max-connections="20"
    min-connections="1">
    <connection-factory
        factory-class="com.oracle.ias.jdbcx.db2.DB2DataSource"
        user="jdoe"
        password="password"
        url="jdbc:oracle:db2://server.foo.com:50000;...>
        <property name="databaseName" value="appdb"/>
        <property name="packageName" value="JDBCPKG"/>
        <property name="serverName" value="server.foo.com"/>
        <property name="portNumber" value="50000"/>
        <xa-recovery-config>
            <password-credential>
                <username>jdoe</username>
                <password>password</password>
            </password-credential>
        </xa-recovery-config>
    </connection-factory>
</connection-pool>

<managed-data-source
    connection-pool-name="ConnectionDB2"
    schema="database-schemas/db2.xml"
    jndi-name="jdbc/OracleDS"
    name="OracleDS"
/>
```

Example 5–2   orion-ejb-jar.xml

```
<enterprise-beans>
    <persistence-manager name="orion"/>
```

```
            <entity-deployment name="EmployeeBean" max-tx-retries="0" location="EmployeeBean">
                <primkey-mapping>
                    <cmp-field-mapping
                        name="empNo" persistence-name="empNo" persistence-type="integer"
                    />
                </primkey-mapping>
                <cmp-field-mapping
                    name="empName" persistence-name="empName"
                />
                <cmp-field-mapping
                    name="salary" persistence-name="salary"
                />
                <finder-method lazy-loading="true">
                    <method>
                        <ejb-name>EmployeeBean</ejb-name>
                        <method-name>findAll</method-name>
                        <method-params></method-params>
                    </method>
                </finder-method>
            </entity-deployment>
</enterprise-beans>
```

Deploying this application will raise an error like:

```
Error creating table: [oias][DB2 JDBC Driver][DB2]ILLEGAL SYMBOL
```

To work around this problem, update the `orion-ejb-jar.xml` to manually define the mapping data types as Example 5–3 shows.

Example 5–3   Updated orion-ejb-jar.xml

```
<enterprise-beans>
    <persistence-manager name="orion"/>
    <entity-deployment name="EmployeeBean" max-tx-retries="0" location="EmployeeBean">
        <primkey-mapping>
            <cmp-field-mapping
                name="empNo" persistence-name="empNo" persistence-type="integer"
            />
        </primkey-mapping>
        <cmp-field-mapping
            name="empName" persistence-name="empName" persistence-type="varchar(255)"
         />
        <cmp-field-mapping
            name="salary" persistence-name="salary" persistence-type="double"
        />
        <finder-method lazy-loading="true">
            <method>
                <ejb-name>EmployeeBean</ejb-name>
                <method-name>findAll</method-name>
                <method-params></method-params>
            </method>
        </finder-method>
    </entity-deployment>
</enterprise-beans>
```

5.3.4 Stateful Session Bean Replication Trigger Configuration

In this release, for stateful session beans, OC4J supports `session-deployment` attribute `replication` settings of:

- `inherited` (default)

- `onShutdown`

- `onRequestEnd`

- `none`

The `replication` attribute for stateful session beans cannot be configured in Application Server Control. The `inherited` value is never displayed and the value cannot be reset to `none`.

To work around this problem, for all stateful session beans, you must manually configure the `orion-ejb-jar.xml` file `session-deployment` element `replication` attribute.

5.3.5 EJB 3.0 Entities and Application Server Control

When you deploy EJB 3.0 entities to OC4J, you cannot manage them using Application Server Control: when you use Application Server Control to view your EJB module, the Entity Beans area will display "No entity beans found".

You can manage all other EJB 3.0 beans such as session beans. For example, if you deploy an EJB module that contains both EJB 3.0 entities and EJB 3.0 session beans, your session beans will be visible through Application Server Control.

5.3.6 Entity and Session Deployment Attribute tx-retry-wait

The `orion-ejb-jar.xml` file `entity-deployment` and `session-deployment` element `tx-retry-wait` attribute is not in `orion-ejb-jar-10_0.xsd` (nor in `orion-ejb-jar.dtd`).

You can still use this attribute in your `orion-ejb-jar.xml` file but if you do, do not configure OC4J to perform XML file validation (using the `-validateXML` option on the OC4J startup command line).

5.4 Release Notes for Web Services

This section describes release notes for Web Services. It covers the following topics:

- Section 5.4.1, "Long File Names Cause Deployment to Fail"
- Section 5.4.2, "SoapFaultException Will Not Invoke a Handler's handleFault Method"
- Section 5.4.3, "Clients Cannot Deserialize SOAP-Encoded anyType Arrays"
- Section 5.4.4, "Arrays in Document-Literal Encoding May Not be Supported when Mapped to a Single Array Parameter"
- Section 5.4.5, "NLS Characters in SYS.XMLTYPE Values May Not be Supported"
- Section 5.4.6, "Self Referential WSDL Imports Fail to Load in the Test Page"
- Section 5.4.7, "SOAP 1.2 Results May Not be Properly Deserialized"
- Section 5.4.8, "WSIF Mapping of Nillable XSD Types"
- Section 5.4.9, "Support for NLS Characters in the WSDL"
- Section 5.4.10, "Multiple Service Elements in Top Down Web Service Assembly"
- Section 5.4.11, "Multiple Message Formats in a WSDL Application"
- Section 5.4.12, "Invalid Configuration Not Detected for EJB 2.1 Web Services"
- Section 5.4.13, "Schema Features Limitations"
- Section 5.4.14, "Limitations on Top Down Processing of Type Mappings"

- Section 5.4.15, "REST-Enabled Web Services Cannot be Deployed with Application Server Control"
- Section 5.4.16, "Explicit HTTP Data Chunking is Not Supported"
- Section 5.4.17, "Runtime Exception Masked By java.io.NotSerializableException"
- Section 5.4.18, "Get NodeLists by Using getFirstChild and getNextSibling Instead of getChildNode"

5.4.1 Long File Names Cause Deployment to Fail

If the combined length of the generated file and directory names passes a certain size limit, then deployment will fail and throw an error. This size limit varies for different operating systems. For example, on the Windows operating system, the size limit is 255 characters. - Bug 4673270

> **Note:** You can avoid this problem by upgrading to a more recent version of the J2SE 5.0 JDK (`jdk-1_5_0_06` or later).

The length of the names is controlled by WebServicesAssembler and the deployment code. WebServicesAssembler generates file names based on the method name in the Java class or the operation name in the WSDL. The deployment code creates directories for code generation based on the names of the EAR and the WAR files.

To avoid the generation of file and directory names that are too long, limit the number of characters in the following names to a reasonable length.

- Method names in Java classes
- Operation names in the WSDL
- Directory name for the location of the OC4J installation (also be aware that the JDeveloper's built-in OC4J instance is typically placed in a directory below the JDeveloper installation)
- File name for a WAR file
- File name for an EAR file

5.4.2 SoapFaultException Will Not Invoke a Handler's handleFault Method

On the server, a `SoapFaultException()` thrown by an implementation class will not invoke a handler's `handleFault()` method. The `handleResponse()` method is called instead.

5.4.3 Clients Cannot Deserialize SOAP-Encoded anyType Arrays

Clients of SOAP-encoded services are not able to deserialize arrays of type `anyType`.

5.4.4 Arrays in Document-Literal Encoding May Not be Supported when Mapped to a Single Array Parameter

Arrays may not be supported in document-literal encoding when mapped directly to a Java method parameter. This issue has been seen in DII and WSIF clients.

It also occurs in document-literal Web services that map `base64Binary` (or `hexBinary`) arrays to the type `byte[][]`.

There are two possible ways to work around this issue:

- Keep the wrapper by specifying the WebServicesAssembler argument `unwrapParam="false"`.
- Use RCP-encoded or RPC-literal styles.

5.4.5 NLS Characters in SYS.XMLTYPE Values May Not be Supported

In Database Web Services, NLS characters that occur in a SQL `SYS.XMLTYPE` value may not be properly handled.

5.4.6 Self Referential WSDL Imports Fail to Load in the Test Page

The test page (Web Services Home Page) fails to load when using self-referential WSDL imports to the same application. For example:

```
location="http://samebox:8888/sameapp/import.wsdl"
```

Since the WSDL is available locally in the application, it should be referenced with a relative path instead. For example:

```
location="./import.wsdl"
```

5.4.7 SOAP 1.2 Results May Not be Properly Deserialized

In certain cases, the SOAP 1.2 response may not be properly deserialized, resulting in an element-name-mismatch exception. Specifically, this happens if the Web services returns output parameters and a result value, but this result element does not immediately follow after an `http://www.w3.org/2003/05/soap-rpc` result element.

5.4.8 WSIF Mapping of Nillable XSD Types

WSIF invocations will map the primitive and nillable XML schema types to primitive Java types. This does not permit the representation of XML nil values.

As a work around, you may want to use SOAP-encoded XML types in the WSDL.

5.4.9 Support for NLS Characters in the WSDL

NLS characters that occur in names in the WSDL, such as in the name of a service, port type, operation, binding or port, are not supported. This may also result in errors on the test page (Web Services Home Page).

5.4.10 Multiple Service Elements in Top Down Web Service Assembly

WebServicesAssembler does not support multiple service elements for the `topDownAssemble` command.

5.4.11 Multiple Message Formats in a WSDL Application

Multiple message formats, such as RPC-encoded and document-literal, are not supported in a single Web application.

5.4.12 Invalid Configuration Not Detected for EJB 2.1 Web Services

EJB 2.1 Web services will be deployed during server side code generation even if the configuration is incorrect.

5.4.13 Schema Features Limitations

This section describes Web Services schema features limitations. It covers the following topic(s):

- Section 5.4.13.1, "Schema Features that are Mapped to a SOAPElement"
- Section 5.4.13.2, "Derived complexTypes Are Not Handled Properly"
- Section 5.4.13.3, "RPC Encoded Does Not Support Complex Types With Attributes"
- Section 5.4.13.4, "XML Types xsd:choice and xsd:group are Not Supported for Proxy or Top Down Web Service Assembly"

5.4.13.1 Schema Features that are Mapped to a SOAPElement

If any of the following schema features are encountered in the WSDL, they will be mapped to a `SOAPElement`.

- Any model group with multiple `xsd:any` elements
- `xsd:choice` elements
- Mixed content
- Substitution groups
- A type with multiple `xsd:anyAttribute`

5.4.13.2 Derived complexTypes Are Not Handled Properly

If a `complexType` derives from another by adding some attributes, then once the complexType is run through the OC4J WSDL2Java tool, all of the attributes in the subtype will be deleted. If the subtype does not have additional elements, it will be presented as a `SOAPElement` in the generated Java code.

If you are able to edit the WSDL, then you can work around this problem in either of the following ways:

- Move the attribute definitions from the sub type to the supertype.
- Avoid using type extensions.

5.4.13.3 RPC Encoded Does Not Support Complex Types With Attributes

If the schema contains a binding with an RPC-encoded message format and WebServicesAssembler encounters a complexType with attributes, then it will throw an "unsupported type encountered" error message.

5.4.13.4 XML Types xsd:choice and xsd:group are Not Supported for Proxy or Top Down Web Service Assembly

If you are assembling Web Services top down or assembling Web service proxies, WebServicesAssembler cannot consume WSDLs that contain the `xsd:choice` or `xsd:group` XML types. If you want to consume a WSDL that contains these XML types, set the WebServicesAssembler `dataBinding` argument to `false` and code the `SOAPElement` so that the payload conforms to the schema definition in the WSDL file.

5.4.14 Limitations on Top Down Processing of Type Mappings

You can specify the WebServicesAssembler `ddFileName` argument to define type mappings. A type mapping maps a schema type to an existing Java class and allows an optional custom serializer. In the top down use case, if you do not supply a custom serializer, then WebServicesAssembler will always generate a bean for the type.

The work around for this limitation is to ensure that the existing Java class is in the classpath given to WebServicesAssembler and that the `overwriteBeans` argument is set to `false`.

5.4.15 REST-Enabled Web Services Cannot be Deployed with Application Server Control

Application Server Control cannot successfully deploy EAR files containing REST-enabled Web services. Instead of using Application Server Control, you can use JDeveloper, Ant, or `admin_client.jar` to deploy the EAR file.

5.4.16 Explicit HTTP Data Chunking is Not Supported

Enabling chunked data transfer for HTTP as described in Chapter 13 of the *Oracle Application Server Web Services Developer's Guide* by explicitly setting `DO_NOT_CHUNK` or `CHUNK_SIZE` properties will not have any effect. However, chunking will still be implicitly enabled when using attachments.

5.4.17 Runtime Exception Masked By java.io.NotSerializableException

When the Web Service client is invoked by an EJB, the RMI protocol requires that the client parameter, return, and exception implement `java.io.Serializable`.

In the current release, however, the `oracle.j2ee.ws.common.util.localization.LocalizableSupport` class does not implement `java.io.Serializable`. Consequently, exceptions thrown by a Web service client invoked by EJB are not properly returned to the invoker. Instead, the invoker receives the description below.

```
Error deserializing return-value: writing aborted;
java.io.NotSerializableException:
/@ oracle.j2ee.ws.common.util.localization.LocalizableSupport; nested /
exception is:
java.io.WriteAbortedException: writing aborted;
java.io.NotSerializableException:
/@ oracle.j2ee.ws.common.util.localization.LocalizableSupport/
java.rmi.UnmarshalException: Error deserializing return-value: writing
aborted; java.io.NotSerializableException:
```

5.4.18 Get NodeLists by Using getFirstChild and getNextSibling Instead of getChildNode

You may see a performance degradation when iterating over a `NodeList` obtained by using `node.getChildNode`. This degradation will only be significant for `NodeLists` with very long lengths.

Instead of using the `NodeList` obtained by `node.getChildNodes`, the current Oracle XDK implementation offers an optimization of navigating a list of child nodes by using `node.getFirstChild` and looping over `node.getNextSibling`. The following code sample illustrates this technique.

```
Node n = ...;
```

```
if (n.hasChildNodes()) {
   for(Node nd=n.getFirstChild(); nd!=null; nd=nd.getNextSibling()){
        nd.getValue(); // do something with nd
      }
}
```

5.5 Release Notes for Web Services Security

This section describes release notes for Web Services Security. It covers the following topic(s):

-

5.5.1 Stale Indirect User Accounts Must be Removed Manually

In release 10.1.3, you must use Application Server Control to obfuscate the keystore, signature key, and encryption key passwords. During obfuscation, an indirect user account is created in the `system-jazn-data.xml` file.

If you undeploy the application, these indirect user accounts are not removed. You must manually delete the them by using Application Server Control.

The following list describes how you can identify the names of indirect user accounts for global-level and port-level keystores and keys.

- For a port-level keystore, the name of the indirect user account is created with the following format:

  applicationName.portName.keystore.*actual-keystore-name*

  For example:

  `my-security-sample.myport.keystore.myks.jks`

- For a global-level keystore, the name of the indirect user account is created with the following format:

  default.keystore.*actual-keystore-name*

  For example:

  `default.keystore.myks.jks`

- For port-level keys, the name of the indirect user account is created with the following format:

  applicationName.portName.key.*actual-key-alias*

  For example:

  `my-security-sample.myport.key.mysignkey`

- For global-level keys, the name of the indirect user account is created with the following format:

  default.key.*actual-key-alias*

  For example:

  `default.key.mysignkey`

5.6 Release Notes for OC4J Services

This section describes release notes for OC4J Services. OC4J Services include: Java Naming and Directory Interface (JNDI), Oracle Enterprise Messaging Service (OEMS), Data Sources, Remote Method Invocation (ORMI and IIOP), OC4J Transaction Support, Java Object Cache (JOC), and XML Query Service (XQS).

The section contains release notes for the following OC4J Services:

- Section 5.6.1, "JNDI"
- Section 5.6.2, "Oracle Enterprise Messaging Service (OEMS)"
- Section 5.6.3, "Data Sources"
- Section 5.6.4, "OC4J Transaction Support"
- Section 5.6.5, "RMI"
- Section 5.6.6, "XQS"

5.6.1 JNDI

This section describes release notes for JNDI. It covers the following topic(s):

- Section 5.6.1.1, "New Package Names for RMI and Application Client Initial Context Factories"
- Section 5.6.1.2, "These Environment Properties Are No Longer Supported"
- Section 5.6.1.3, "Context Factory Restructuring"
- Section 5.6.1.4, "Objects that Implement javax.naming.Referenceable Interface"

5.6.1.1 New Package Names for RMI and Application Client Initial Context Factories

In this release, note the following new package names for the initial context factories:

- `oracle.j2ee.rmi.RMIInitialContextFactory`
- `oracle.j2ee.naming.ApplicationClientInitialContextFactory`

5.6.1.2 These Environment Properties Are No Longer Supported

The following environment properties are no longer supported as of release 10.1.3:

`dedicated.connection`

`dedicated.rmicontext`

In release 10.1.3, the known ORMI /JNDI bugs that required these flags have been resolved. To enable client-side ORMI load-balancing in 10.1.3, use the `oracle.j2ee.rmi.loadBalance` property described in the "Load Balancing" section of the JNDI chapter of the *Oracle Containers for J2EE Services Guide*.

5.6.1.3 Context Factory Restructuring

The package structure for context factories provided by previous releases of OC4J is deprecated, and is replaced by a more consistent naming structure. The following context factories are deprecated in release 10.1.3:

- `com.evermind.server.rmi.RMIInitialContextFactory`
- `com.evermind.server.ApplicationClientInitialContext Factory`
- `com.oracle.iiop.server.IIOPInitialContextFactory`

For the new context factory names that replace the deprecated ones, see the `java.naming.factory.initial` initial context property described in the "Constructing a JNDI Context" section of the JNDI chapter of the *Oracle Containers for J2EE Services Guide*.

5.6.1.4 Objects that Implement javax.naming.Referenceable Interface

OC4J JNDI in 10.1.3 now provides full support for binding objects that implement the `javax.naming.Referenceable` interface

5.6.2 Oracle Enterprise Messaging Service (OEMS)

This section describes release notes for the Oracle Enterprise Messaging Service (OEMS). It covers the following topic(s):

- Section 5.6.2.1, "Special Considerations For Undeploying the Default Instance of the Oracle gJRA Resource Adapter"

- Section 5.6.2.2, "OC4J May Fail to Restart after Abnormal OC4J Shutdown"

- Section 5.6.2.3, "getconfigProperties() Lists Some Unsupported Properties"

5.6.2.1 Special Considerations For Undeploying the Default Instance of the Oracle gJRA Resource Adapter

OC4J cannot be started with OracleASjms, the pre-packaged standalone JMS Connector, undeployed without certain changes. This note deals with additional changes necessary to start OC4J while the default instance of the Oracle gJRA resource adapter, OracleASjms, is undeployed. For general undeployment of a resource adapter, see the Oracle Containers for J2EE Services Guide. The following additional changes must be made:

- In `$J2EE_HOME/config/application.xml` comment out the following lines:

  ```
  <web-module id="jmsrouter_web" path="../../home/applications/jmsrouter.war" />
  <ejb-module id="jmsrouter_ejb" path="../../home/applications/jmsrouter-ejb.jar"
  />
  ```

- In `$J2EE_HOME/config/default-web-site.xml`, comment out the following line

  ```
    <web-app application="default" name="jmsrouter_web" root="/jmsrouter"
  load-on-startup="true" />
  ```

If these changes are made, OC4J may be started, but the OracleAS JMS Router will not work.

To reinstate the JMS Router:

1. Fully redeploy the OracleASjms resource adapter instance.

2. Uncomment the lines mentioned above in
   `$J2EE_HOME/config/application.xml`
   and
   `$J2EE_HOME/config/default-web-site.xml`.

When OC4J is restarted, the OracleAS JMS Router should be available.

5.6.2.2 OC4J May Fail to Restart after Abnormal OC4J Shutdown

If you encounter OC4J JMS Server startup problems after an abnormalOC4J shutdown, first check that no other OC4J JMS Server is running and using the same persistence files. Then remove any.lock files from the

```
ORACLE_HOME/j2ee/instance_name/persistence
```

directory and try restarting again.

If problems persist, confirm that the jms.xml file is valid.

If problems still persist, remove the jms.state file from the persistence directory and try again, but be aware that removing this file may result in loss of transaction information. For additional information, see the section "Abnormal Termination" in the "Oracle Enterprise Messaging Service" chapter of the *Oracle Containers for J2EE Services Guide*.

5.6.2.3 getconfigProperties() Lists Some Unsupported Properties

The list of properties returned by the JMS Administrator MBean's getconfigProperties() method includes the following properties that are neither documented nor supported:

- oc4j.jms.checkPermissions
- oc4j.jms.j2ee14
- oc4j.jms.noJmx
- oc4j.jms.printStackTrace
- oc4j.jms.rememberAllXids

5.6.3 Data Sources

This section describes release notes for Data Sources. It covers the following topics:

- Section 5.6.3.1, "New Syntax for Data Source Configuration"
- Section 5.6.3.2, "OracleConnectionCacheImpl Deprecated"
- Section 5.6.3.3, "Converting Existing Data Sources to Release 3 Format"
- Section 5.6.3.4, "Configuring OC4J instance for oci drivers"

5.6.3.1 New Syntax for Data Source Configuration

The data sources subsystem has been completely rewritten. Part of the rewrite includes a new syntax for the configuration using the data-sources.xml file. The pre-10.1.3 syntax is still supported but users are encouraged to use the new syntax. Also users are encouraged to convert their existing application data-sources.xml files to the new syntax. See the *Oracle Containers for J2EE Services Guide* for details on converting data-sources.xml.

5.6.3.2 OracleConnectionCacheImpl Deprecated

The class oracle.jdbc.pool.OracleConnectionCacheImpl has been deprecated because it does not support multiple schemas. When defining the factory-class for connection factories and data-source-class for native data sources, use oracle.jdbc.pool.OracleDataSource instead of oracle.jdbc.pool.OracleConnectionCacheImpl.

5.6.3.3 Converting Existing Data Sources to Release 3 Format

The OC4J 10.1.3 implementation understands the 10.1.3 and the pre-10.1.3 (10.1.2 and 9.0.4) formats of the `data-sources.xml` file. For an application that was previously used in a pre-10.1.3 OC4J implementation and contains its own `data-sources.xml` file, the OC4J 10.1.3 implementation automatically converts the `data-sources.xml` file from the pre-10.1.3 format to the 10.1.3 format when you use the Application Server Control Console to change anything in the `data-sources.xml` file, such as modifying an existing data source or creating or deleting a data source.

With an active OC4J instance in a standalone environment, you can alternatively use `admin.jar` with the following syntax to manually convert a pre-10.1.3 `data-sources.xml` file to the 10.1.3 format. This is discussed in the *Oracle Containers for J2EE Configuration and Administration Guide*. The guide does not mention that you can specify an ORMI URL only when OC4J is running or that the ORMI URL is optional.

```
java -jar admin.jar ormi://oc4jHost:oc4jOrmiPort adminId adminPassword
-convertDataSourceConfiguration old-data-sources.xml new-data-sources.xml
```

You can also convert a `data-sources.xml` file before deployment, without a running OC4J instance. The syntax for this offline conversion is as follows:

```
java -jar admin.jar -convertDataSourceConfiguration old-data-sources.xml
new-data-sources.xml
```

Notes:

- If you include the ORMI port, then OC4J must be running. When OC4J is not running, you must omit the ORMI URL from the admin.jar command line.

- If you do not include the ORMI port, then the admin.jar command will work either way, that is, with OC4J running or with OC4J not running.

- The admin.jar utility works only in the standalone OC4J environment. This utility is installed in the Oracle Application Server environment, but does not work in an OPMN-managed environment.

- The newer admin_client.jar utility works in both environments, standalone and managed Oracle Application Server. However, the admin_client.jar utility does not convert data-sources.xml files.

Check for Consistency Between Your Application and the New data-sources.xml File

After conversion, whether manual or automatic, visually inspect the new `data-sources.xml` file and confirm that there is consistency between your application and the new file regarding the JNDI location used to refer to a data source. This is advisable because the new file may contain data source definitions that are not used.

This happens because the old format uses multiple location attributes (such as `location`, `ejb-location`, `xa-location`, and so on). The conversion to the new 10.1.3 format creates a separate data source in the new `data-sources.xml` file corresponding to each location attribute specified in the old `data-sources.xml` file. In most cases, client applications will only use the data source defined by either the

location or ejb-location attribute. But we cannot be sure of this. Therefore, the converted data-sources.xml may have definitions that are not used by the applications and can be removed from the file.

You can also refer to examples of the new data-sources.xml format in the "Data Sources" chapter of the *Oracle Containers for J2EE Services Guide*.

5.6.3.4 Configuring OC4J instance for oci drivers

If any application uses oci drivers for a datasource connection then, add the following *<environment>* section in opmn.xml for that particular OC4J instance. For example, if the application is deployed to the default home instance, then configure the OC4J instance as follows:

```
<ias-component id="OC4J">
   <environment>
      <variable id="SHLIB_PATH" value="$ORACLE_HOME/lib32" />
   </environment>
     <process-type id="home" module-id="OC4J" status="enabled">
```

5.6.4 OC4J Transaction Support

This section describes release notes for OC4J Transaction Support. It covers the following topics:

- Section 5.6.4.1, "Change the Default JTA Recovery Password Immediately"

- Section 5.6.4.2, "New Configuration File for Transaction Manager"

- Section 5.6.4.3, "The In-DB Coordinator Is Deprecated"

- Section 5.6.4.4, "The Mid-Tier Coordinator Does Not Use a Persistent Store By Default"

- Section 5.6.4.5, "DMS must be enabled to obtain JTA statistics"

- Section 5.6.4.6, "Transaction Propagation Between 10.1.3 Instances Only"

5.6.4.1 Change the Default JTA Recovery Password Immediately

The default JTA recovery password should be changed immediately. OC4J is shipped with a default password, which should be changed after install. The recovery password is configured in the configuration file jazn-data.xml, which is in the $J2EE_HOME/config directory. To modify the transaction recovery password, change the credentials value for the user JtaAdmin in the jazn-data.xml file.

```
<user>
    <name>JtaAdmin</name>
    <display-name>JTA Recovery User</display-name>
    <description>Used to recover propagated OC4J transactions</description>
    <credentials>!newJtapassword</credentials>
</user>
```

Even if OC4J is configured to use a security service other than JAZN, such as OID the transaction recovery password must still be configured in jazn-data.xml.

5.6.4.2 New Configuration File for Transaction Manager

All transaction-manager-related configuration is now done in the transaction-manager.xmlfile.

5.6.4.3 The In-DB Coordinator Is Deprecated

The use of the in-database transaction coordinator by OC4J is deprecated as of release 10.1.3. Oracle recommends that the middle-tier transaction coordinator be used going forward.

5.6.4.4 The Mid-Tier Coordinator Does Not Use a Persistent Store By Default

The mid-tier coordinator does not use a persistent store by default. Prior to use in production, the mid-tier coordinator should be configured to use a persistent store which will enable transaction recovery.

5.6.4.5 DMS must be enabled to obtain JTA statistics

To obtain JTA statistics, ensure that DMS is enabled.

5.6.4.6 Transaction Propagation Between 10.1.3 Instances Only

Transaction propagation is only supported between 10.1.3 instances.

5.6.5 RMI

This section describes release notes for OC4J Remote Method Invocation (RMI and IIOP). It covers the following topics:

- Section 5.6.5.1, "RMI Recommendations"

- Section 5.6.5.2, "Excessive ORMI Connections Created"

- Section 5.6.5.3, "Workaround for HTTP Tunnelling Failover"

- Section 5.6.5.4, "Incorrect "Provider URL..." Error Message"

5.6.5.1 RMI Recommendations

In this release, note the following recommendations:

- Environment variables `dedicated.connection` and `dedicated.rmicontext` are not required for lookup using EJBs in 10.1.3

  The `dedicated.connection` environment variable is still required for EJBs hosted in 10.1.2 being looked up from 10.1.3 - Bug 4895256

- The RMI port may not be released immediately sometimes. - Bug 4892487

- Old tunneling is deprecated. Use the new URL, as described in the "Configuring ORMI Tunneling through HTTP" section of the "RMI" chapter of the *Oracle Containers for J2EE Services Guide*.

- Old package names for context factories deprecated and new names are recommended to be used, as described in the "Constructing JNDI Context" section of the "JNDI" chapter of the *Oracle Containers for J2EE Services Guide*.

5.6.5.2 Excessive ORMI Connections Created

Setting the JNDI property `oracle.j2ee.rmi.loadBalance` to either `context` or `lookup` currently creates separate ORMI connections for each call to new `InitialContext`. Closing the context does not cause the connection to be closed. Doing this repeatedly will result in performance degredation. - Bug 4902304

5.6.5.3 Workaround for HTTP Tunnelling Failover

The following workaround is necessary for HTTP Tunnelling failover to work in iAS mode. This workaround applies to iAS mode, where the provider URL points to OHS using `mod_oc4j`. The workaround is not needed in standalone mode, where the provider URL lists multiple OC4J instances.- Bug 4599521

The workaround must be done for all OC4J server instances in the cluster, such as home1, home2, and so on.

1. Go to Application Server Control Console.

2. Select a server instance, such as "home".

3. In the instance window, select the Applications tab.

4. Select the "default" application.

5. In the Application:default window, select the Administration tab.

6. Select the Clustering Properties task.

7. Select the "Override parent application clustering settings" radio button. Specify Clustering Enable.

8. Click the OK button.

9. Repeat for each OC4J server instance.

10. Add the `<distributable />` element to the `<ORACLE_HOME>/j2ee/home/default-web-app/WEB-INF/web.xml` file.

11. Restart the server using `opmnctl stopall` and then `opmnctl startall`.

5.6.5.4 Incorrect "Provider URL..." Error Message

In certain cases when there is something wrong with the provider url format, the following incorrect error message is displayed:

```
" Provider URL must be of the form
[opmn:]corbaname::host:port#/appname"
```

The URL format in the error message is incorrect. The correct URL format is:

```
 [opmn:]corbaname::host:port#[instancename#]appname
```

5.6.6 XQS

This section describes release notes for XML Query Service (XQS). It covers the following topic(s):

- Section 5.6.6.1, "Implementation Restriction on the fn:doc() and fn:collection() Functions"

5.6.6.1 Implementation Restriction on the fn:doc() and fn:collection() Functions

The only arguments that the current implementation of the built-in XQuery functions `fn:doc` and `fn:collection` support are URLs with the "file" protocol that specify a path to a file on the local file system, as in this function call:

```
 fn:doc("C:/MyDocuments/XQS/myView.xq")
```

The protocol part of the URL is always assumed to be "file" and can be omitted.

As an alternative to the `fn:doc()` function, you can use an XQS document function to access a document using any URL that Oracle Application Server supports. For example, a `<document-source>` element has the following configuration:

```
<document-source>
        <function-name prefix="ns"> genericFile </function-name>
</document-source>
```

The XQS function `genericFile()` can be used in a query expression, as follows:

```
declare namespace ns = "…";
declare function ns:genericFile() external;
```

5.7 Release Notes for J2EE Connector Architecture (J2CA)

This section describes release notes for J2EE Connector Architecture (J2CA). It covers the following topics:

- Section 5.7.1, "J2CA Lifecycle Issues"

- Section 5.7.2, "Cannot Cast a Connection Handle to a Concrete Type"

- Section 5.7.3, "RAR Name Must Be Unique"

- Section 5.7.4, "Set inactivity-timeout-check in oc4j-ra.xml"

- Section 5.7.5, "Stop the Resource Adapter Before Redeploying It"

- Section 5.7.6, "Explicit Configuration Is Necessary For Resource Adapter To Support XA Transaction Recovery"

- Section 5.7.7, "ASControl Changes to Work Manager Thread Pool Not Persisted If <work-manager-thread-pool> Not Defined"

5.7.1 J2CA Lifecycle Issues

- Unable to deploy multiple versions of a standalone RAR - Bug 4253861

- A standalone RAR takes precedence over one in an application. When the same fully-qualified class exists in both a standalone RAR and also in a RAR deployed in an EAR, the class will always be loaded from the standalone RAR. - Bug 4415389

- When stopping a resource adapter, OC4J does not always properly stop dependent applications.

5.7.2 Cannot Cast a Connection Handle to a Concrete Type

OC4J wraps all connection handles with connection handle proxies to perform connection association and therefore connection handles can only be cast to interfaces implemented by the connection handle. An attempt to cast a connection handle to a concrete class will cause a `ClassCastException`.

5.7.3 RAR Name Must Be Unique

`NullPointerException` occurs if an attempt is made to deploy an RAR when there is already an RAR deployed with the same name. - Bug 4884317

5.7.4 Set inactivity-timeout-check in oc4j-ra.xml

Set `inactivity-timeout-check` in the `oc4j-ra.xml` file. Changing the `inactivity-timeout-check` property for an RAR connection pool with ASControl does not work properly. This property should be set to the proper value in the `oc4j-ra.xml` file prior to deploying the resource adapter. - Bug 4455421

5.7.5 Stop the Resource Adapter Before Redeploying It

When a resource adapter with active endpoints is redeployed without stopping it first, OC4J throws a `DeployerException` due to active endpoints. To work around this issue, stop the resource adapter prior to redeploying it. - Bug 4740441

5.7.6 Explicit Configuration Is Necessary For Resource Adapter To Support XA Transaction Recovery

XA transaction recovery can be configured using Application Server Control using the following steps:

1. In the `Connection Factories` tab accessed from the `Resource Adapter Home` page for the appropriate resource adapter, choose the JNDI location of the connection factory that you want to configure.

2. In the `Options` tab of the resulting `Edit Connection Factory` page, you can do any of the following:

   ■ Add a new user name.

     After specifying the user name, you can specify a password directly or indirectly. For a direct password, choose `Password` and type the password itself.

     For an indirect password, choose `Indirect Password` and type a key (which might just be the user name, for example). OC4J uses the key to do a lookup in the User Manager (specifically, in the jazn-data.xml file).

   ■ Change an existing user name or password

5.7.7 ASControl Changes to Work Manager Thread Pool Not Persisted If <work-manager-thread-pool> Not Defined

Changes to work manager thread pool properties from ASControl are not persisted to the `server.xml` file if there is no `<work-manager-thread-pool>` element defined. - Bug 4871940

5.8 Release Notes for OracleAS JAAS Provider and Security

This section describes release notes for the OracleAS JAAS Provider in Release 10.1.3.0.0. It covers the following topics:

■ Section 5.8.1, "COREid Status for 10.1.3.0.0"

■ Section 5.8.2, "Restart Application After Configuring Through Security Provider MBean"

■ Section 5.8.3, "Necessary Permission Grants When Using Security Manager"

■ Section 5.8.4, "Indirect Users for Password Indirection"

■ Section 5.8.5, "JAAS Policy Configuration with Custom Realms"

- Section 5.8.6, "User Manager Delegation for the File-Based Provider"
- Section 5.8.7, "JNDI Context Pool Timeout Property for Oracle Internet Directory"
- Section 5.8.8, "Miscellaneous OracleAS JAAS Provider and Security Release Notes"

5.8.1 COREid Status for 10.1.3.0.0

The initial version of the 10.1.3.0.0 OC4J Release Notes pointed out that as of the 10.1.3.0.0 release, you cannot use the COREid Access security provider for J2EE Web applications deployed in 10.1.3 OC4J. This update is to point out that, in fact, you cannot yet use any functionality through the COREid custom login module. In other words, COREid integration with Web and EJB applications will not be supported until a patch is made available. (Because the Oracle Web Services Manager agent covers integration with COREid, there *is* COREid integration for Web services in the 10.1.3.0.0 implementation.) Refer to OracleMetaLink to check the status of the future 10.1.3.0.0 patch set. - Bugs 4887466, 4772443, 4745790

5.8.2 Restart Application After Configuring Through Security Provider MBean

Whenever a configuration change is made using Application Server Control or the OC4J security provider MBean, the application must be restarted. Until the application is restarted, all other operations of the security provider MBean are invalidated and will return the following message: "The security provider has been changed. Operation temporarily invalidated till application or OC4J restart."

5.8.3 Necessary Permission Grants When Using Security Manager

Users running with a `SecurityManager` in an Oracle Application Server environment should be aware that if an OC4J instance name other than `home` is used, adding the following permission grants to *ORACLE_HOME*/j2ee/*instance_name*/config/java2.policy will be necessary for proper operation of OC4J:

```
grant codebase
"file:${oracle.home}/j2ee/${oracle.oc4j.instancename}/connectors/OracleASjms/Oracl
eASjms/gjra.jar" {
    permission java.security.AllPermission;
};

grant codebase "file:
${oracle.home}/j2ee/${oracle.oc4j.instancename}/connectors/datasources/datasources
/datasources.jar"
{
    permission java.security.AllPermission;
};
```

(Failure to add these does not compromise security but may hinder OC4J operations.) - Bug 4942880

5.8.4 Indirect Users for Password Indirection

If you choose to use indirect passwords in the OC4J 10.1.3.0.0 implementation, an indirect user is created in the `system-jazn-data.xml` file when you use this feature. Be aware that these indirect user accounts are not removed automatically when an application is undeployed; you must use Application Server Control Console to delete any stale indirect user accounts manually.

5.8.5 JAAS Policy Configuration with Custom Realms

When you use custom realms, and JAAS policies are granted to users or roles in the custom realm, you should do the following:

1. In the `<jazn>` element of your application `orion-application.xml` file, specify a `default-realm` setting of "*custom_realm_name*".

2. Do *not* specify a `location` attribute setting in the `<jazn>` element.

3. Set the `jaas.username.simple` property to "`false`" in `jazn.xml`, using a `<property>` subelement of the `<jazn>` element.

These steps allow the custom realm and its users, roles, and policies to be persisted in `system-jazn-data.xml`.

Note that to use JAAS authorization, in particular to grant permissions to users or roles in a custom realm, the custom realm and its users and groups must be defined and persisted in `system-jazn-data.xml`, not in a `jazn-data.xml` file deployed in the application EAR file.

5.8.6 User Manager Delegation for the File-Based Provider

Before HTTP requests can be dispatched to the target servlet, the OracleAS JAAS Provider `JAZNUserManager` coordinates authentication. `JAZNUserManager` supports the OC4J `UserManager` delegation model, but effectively this applies only to the file-based provider. With delegation, if a user or group is not found at the application-level `JAZNUserManager` instance, the request is delegated to the parent user manager.

Specifically, note the following restrictions and additional details:

- If the application and parent application are both configured to use the file-based provider, delegation goes up through the parent hierarchy as far as necessary, until a parent is not configured to use the file-based provider. Delegation is not propagated beyond that point.

- If the application is configured to use the file-based provider, and the parent is configured to use the LDAP-based provider, an external LDAP provider, or a custom login module, there is no delegation support.

- If the application itself is configured to use the LDAP-based provider, an external LDAP provider, or a custom login module, there is no delegation support.

> **Note:** In OC4J, the `system` application is at the root of the hierarchy, but the `default` application is the default parent of any deployed application. Both use `system-jazn-data.xml` as the user repository.

5.8.7 JNDI Context Pool Timeout Property for Oracle Internet Directory

For the LDAP-based provider (Oracle Internet Directory), the OC4J 10.1.3 implementation includes a new property, `JNDI_CTX_POOL_TIMEOUT`, that you can set in order to specify a timeout for the JNDI context pool. This may be useful, for example, when there is a firewall between the middle tier, including OracleAS JAAS Provider, and the Oracle Internet Directory. The timeout on the firewall connection could be coordinated with the timeout of the directory context.

Set this property through a `<property>` subelement of the `<jazn>` element in the `jazn.xml` file, specifying the timeout in milliseconds. The following example specifies a timeout of 5 seconds.

```
<jazn ... >
   <property name="JNDI_CTX_POOL_TIMEOUT" value="5000">
   ...
</jazn>
```

5.8.8 Miscellaneous OracleAS JAAS Provider and Security Release Notes

- Although it is already stated in the 10.1.3.0.0 OC4J Security Guide, make special note of the fact that the OC4J administration account, `admin` in previous releases, has changed to `oc4jadmin`.

- Security context propagation is supported only between OC4J 10.1.3 instances.

5.9 Release Notes for Documentation Errata

This section describes known errors in the OC4J documentation in Oracle Application Server 10g Release 3 (10.1.3). It covers the following book(s):

- Section 5.9.1, "Web Services Documentation Errata"

- Section 5.9.2, "Oracle Application Server Advanced Web Services Developer's Guide Documentation Errata"

- Section 5.9.3, "Oracle Containers for J2EE Services Guide"

5.9.1 Web Services Documentation Errata

This section describes Web Services documentation errata. It covers the following topic(s):

- Section 5.9.1.1, "WebServicesAssembler Command genInterface Does Not Use the use and style Arguments"

- Section 5.9.1.2, "Error in Ant Task for Assembling JMS Web Services"

5.9.1.1 WebServicesAssembler Command genInterface Does Not Use the use and style Arguments

Book: *Oracle Application Server Web Services Developer's Guide*

Chapter 4, "OracleAS Web Services message Formats", Section: "Selecting Message Formats"

The list of WebServicesAssembler commands that can use the `use` and `style` arguments includes `genInterface`. This is an error. The `genInterface` command cannot use these arguments.

5.9.1.2 Error in Ant Task for Assembling JMS Web Services

Book: *Oracle Application Server Web Services Advanced Developer's Guide*

Chapter 8, "Using JMS as a Web Service Transport", Section: "Assembling a Web Service Bottom Up that Uses JMS Transport"

In Step 1, there is a missing closing angle bracket in the `<oracle:porttype` clause, at the end of the `sendConnectionFactoryLocation` attribute. The `<oracle:porttype` clause should read as follows:

```
...
   <oracle:porttype
      interfaceName="oracle.j2ee.ws.jmstransport.Echo"
      className="oracle.j2ee.ws.jmstransport.EchoImpl"
      >
      <oracle:port
         uri="/echo"
         sendQueueLocation="jms/senderQueue"
         name="EchoPort"
         sendConnectionFactoryLocation="jms/senderQueueConnectionFactory">
      </oracle:port>
...
```

5.9.2 Oracle Application Server Advanced Web Services Developer's Guide Documentation Errata

This section describes errors in the *Oracle Application Server Advanced Web Services Developer's Guide*. It covers the following item(s):

- Section 5.9.2.1, "Auditing and Logging File Path Corrections"

5.9.2.1 Auditing and Logging File Path Corrections

The *Oracle Application Server Advanced Web Services Developer's Guide* Chapter 6, "Auditing and Logging Messages" lists the paths to the auditing and logging `log.xml` files as:
```
<ORACLE_HOME>\log\wsmgmt\logging\log.xml
```
and
```
<ORACLE_HOME>\log\wsmgmt\audit\log.xml
```

These paths are incorrect. The correct paths are:
```
<ORACLE_HOME>\j2ee\<OC4J_instance_
name>\log\wsmgmt\auditing\log.xml
```
and
```
<ORACLE_HOME>\j2ee\<OC4J_instance_
name>\log\wsmgmt\logging\log.xml
```

5.9.3 Oracle Containers for J2EE Services Guide

This section describes errors in the *Oracle Containers for J2EE Services Guide*. It covers the following item(s):

- Section 5.9.3.1, "Incorrect URL in Native Data Source Example for Fast Connection Failover"

5.9.3.1 Incorrect URL in Native Data Source Example for Fast Connection Failover

In the *Oracle Containers for J2EE Services Guide* Chapter 4, "Data Sources", the "Enabling Fast Connection Failover in the data-sources.xml File" section, in the native data source example for fast connection failover labelled "Enabling Fast Connection Failover in the data-sources.xml File" is incorrect.

- The url is not correct for a RAC environment.

- The brackets in the <native-data-source> element do not match correctly.

The **INCORRECT** example is as follows:

```
<native-data-source>
```

```
  name="nativeDataSource"
  jndi-name="jdbc/nativeDS"
  description="Native DataSource"
  data-source-class="oracle.jdbc.pool.OracleDataSource"
  user="scott"
  password="tiger"
  url="jdbc:oracle:thin:@localhost:1521:oracle">
 <property name="connectionCacheName" value="ICC1"/>
 <property name="connectionCachingEnabled" value="true"/>
 <property name="fastConnectionFailoverEnabled" value="false"/>
</native-data-source>
```

A **CORRECT** example is as follows:

```
<native-data-source
  name="nativeDataSource"
  jndi-name="jdbc/nativeDS"
  description="Native DataSource"
  data-source-class="oracle.jdbc.pool.OracleDataSource"
  user="scott"
  password="tiger"

url="jdbc:oracle:thin:@(DESCRIPTION=
  (LOAD_BALANCE=on)
  (ADDRESS=(PROTOCOL=TCP)(HOST=host1) (PORT=1521))
  (ADDRESS=(PROTOCOL=TCP)(HOST=host2)(PORT=1521))
  (CONNECT_DATA=(SERVICE_NAME=service_name)))">

 <property name="connectionCacheName" value="ICC1"/>
 <property name="connectionCachingEnabled" value="true"/>
 <property name="fastConnectionFailoverEnabled" value="false"/>
</native-data-source>
```

5.10 Oracle Application Server Containers for J2EE Job Scheduler

This section describes issues associated with Oracle Application Server Containers for J2EE Job Scheduler. It includes the following topics:

- Section 5.10.1, "Invalid Data Source Configuration May Result in Initialization Exception"

- Section 5.10.2, "Cancel API is not Transactional"

- Section 5.10.3, "Lower Than Expected Throughput may be Experienced for Large Number of Jobs"

- Section 5.10.4, "Removing a Job May Impact Job Scheduler Event Listener Processing"

- Section 5.10.5, "Preemptory Shutdown of OC4J Container may Prevent Subsequent Restart"

5.10.1 Invalid Data Source Configuration May Result in Initialization Exception

In the JDBC persistence configuration, a null pointer exception results on container startup if the associated data source is improperly configured or the database server is not up.

There is no workaround for this issue; make sure the data sources are configured correctly and the target database is up.

5.10.2 Cancel API is not Transactional

If the Cancel API is invoked within a JTA transaction, all outstanding executions are canceled synchronously, not after the transaction is committed.

There is no workaround for this issue.

5.10.3 Lower Than Expected Throughput may be Experienced for Large Number of Jobs

Lower than expected execution throughput may be observed when there are large burst jobs with concurrent schedules.

To work around this issue, disable the management bean and DMS statistics publication in order to increase throughput. This can be accomplished by setting the value of the following environment entries in the Job Scheduler configuration to false:

- `oracle.ias.scheduler.dms`

- `oracle.ias.scheduler.jmx`

5.10.4 Removing a Job May Impact Job Scheduler Event Listener Processing

Removing a job in the JMS persistence configuration may result in event processing delays. This behavior is exacerbated in a deployment where jobs are created and removed with high frequency.

To work around this issue, disable the management bean and DMS statistics publication in order to increase throughput. This can be accomplished by setting the value of the following environment entries in the Job Scheduler configuration to false:

- `oracle.ias.scheduler.dms`

- `oracle.ias.scheduler.jmx`

5.10.5 Preemptory Shutdown of OC4J Container may Prevent Subsequent Restart

The JMS server creates recover lock files in the *$ORACLE_HOME*/j2ee/home/persistence directory. As a result of a preemptory shutdown, these files may not be properly cleaned up and may prevent the container from restarting. Refer to the JMS release notes for more information.

This issue is only pertinent to Job Scheduler running in a JMS persistence environment.

6

Oracle Application Server Portal

This chapter describes issues associated with OracleAS Portal. It includes the following topic:

- Section 6.1, "General Issues and Workarounds"
- Section 6.2, "Portlet and Provider Issues and Workarounds"
- Section 6.3, "Documentation Errors"

6.1 General Issues and Workarounds

This section describes general issues and workarounds. It includes the following topic:

- Section 6.1.1, "Ensure Correct JDK Version in Compiler Settings"

6.1.1 Ensure Correct JDK Version in Compiler Settings

Java classes compiled using JDK 1.5 will not run in JDK 1.4. The default JDK version used to compile code depends on the JDK version used by the Oracle JDeveloper IDE.

If you create a portlet using Oracle JDeveloper 10*g* Release 3 (10.1.3) running on JDK 1.5, and plan to deploy it to an OC4J instance that uses JDK 1.4, then you must ensure that the compiler settings are set to 1.4. To set the project compiler settings, perform the following steps:

1. In Oracle JDeveloper, right-click your project in the Applications Navigator tab, and select **Project Properties** from the context menu.

2. From the Project Properties dialog box, click **Compiler** in the left pane.

3. From the **Source** list, select **1.4**.

4. Click **OK**.

6.2 Portlet and Provider Issues and Workarounds

This section describes issues and workarounds related to OmniPortlet, Web Clipping, Simple Parameter Form, Page portlet, and WSRP providers. This section includes the following topic:

- Section 6.2.1, "Error Configuring OC4J Standalone for JPS Portlets Deployment"

6.2.1 Error Configuring OC4J Standalone for JPS Portlets Deployment

As part of the procedure to configure an OC4J standalone instance for deploying Java Portal Services (JPS) portlets, you are required to create a database preference store to

store the JPS portlets registration and preference information. This includes creating the preference store using the `ptlwsrp_data.sql` script, and editing the `data-sources.xml` file to add a new entry that maps the connection details for the preference store schema to a JDBC data source.

If you installed OC4J release 10.1.3, and added a `data-source` entry in the `data-sources.xml` file, then you will get errors. This is because data sources are defined differently in OC4J release 10.1.3 and the format of the `data-sources.xml` file has changed. You must add a new entry in the `data-sources.xml` file as shown in the following example:

```
<managed-data-source connection-pool-name='Connection Pool'
jndi-name='jdbc/portletPrefs' name='PortletPrefs' />
<connection-pool name='Connection Pool' >
  <connection-factory factory-class='oracle.jdbc.pool.OracleDataSource'
    user='portlet_prefs' password='portlet_prefs'
    url="jdbc:oracle:thin:@(description=(address=
      (host=myhost.domain.com)(protocol=tcp)(port=1521))
      (connect_data=(service_name=myservice.mydomain.com)))">
  </connection-factory>
</connection-pool>
```

6.3 Documentation Errors

This section describes known errors in OracleAS Portal documentation. It includes the following topic:

- Section 6.3.1, "Configuring OC4J Standalone"

6.3.1 Configuring OC4J Standalone

In *Oracle Application Server Portal Developer's Guide*, step 7 under Section 6.3.2, "Configuring OC4J Standalone" is incorrect. This step should read as follows:

Edit the file `data-sources.xml` located in *OC4J_HOME*/j2ee/home/config, where *OC4J_HOME* is the location in which you unzipped OC4J. Add a new entry that maps the connection details for the preference store schema to a JDBC data source with an ejb-location of `jdbc/portletPrefs`. For example:

```
<managed-data-source connection-pool-name='Connection Pool'
jndi-name='jdbc/portletPrefs' name='PortletPrefs' />
<connection-pool name='Connection Pool' >
  <connection-factory factory-class='oracle.jdbc.pool.OracleDataSource'
    user='portlet_prefs' password='portlet_prefs'
    url="jdbc:oracle:thin:@(description=(address=
      (host=myhost.domain.com)(protocol=tcp)(port=1521))
      (connect_data=(service_name=myservice.mydomain.com)))">
  </connection-factory>
</connection-pool>
```

7

Oracle TopLink

This chapter describes issues associated with Oracle TopLink (TopLink). It includes the following topic:

- Section 7.1, "General Issues and Workarounds"

7.1 General Issues and Workarounds

This section describes general issue and workaround. It includes the following topic:

- Section 7.1.1, "Links to Hosted Documentation and Web-Based Resources from TopLink Workbench"
- Section 7.1.2, "Using Non-ASCII Characters with a JAXB 1.0 TopLink Project"
- Section 7.1.3, "TopLink Workbench Look and Feel With Linux GTK"
- Section 7.1.4, "Unit of Work and JTA Transactions"
- Section 7.1.5, "Using JDBC/ODBC Driver"

7.1.1 Links to Hosted Documentation and Web-Based Resources from TopLink Workbench

When using the online help and the hosted documentation, TopLink Workbench requires an internet connection to OTN (Oracle Technology Network). Your PC must have a network connection (including any required proxy information) and an internet browser for use with the TopLink Workbench. If the TopLink Workbench is unable to connect to OTN, some links in the online help or the Welcome page may not work properly.

To configure preferences for an external browser for use with the TopLink Workbench, select **Tools** > **Preferences** > **Help**. Click **Browse** and select the location of your default Web browser. For more information on Help preferences, go to the Help menu and select *Working With TopLink Workbench Preferences*.

7.1.2 Using Non-ASCII Characters with a JAXB 1.0 TopLink Project

When you generate class and method names that include non-ASCII characters, TopLink will throw an exception. This problem occurs when creating a JAXB 1.0 TopLink project from an XML schema that contains non-ASCII characters. Ensure that your XML schema does not contain any non-ASCII characters.

7.1.3 TopLink Workbench Look and Feel With Linux GTK

When using TopLink Workbench on a PC using the Linux GTK look and feel, do not set the TopLink Workbench Look and Feel Preference to **GTK+**. Doing so may cause unexpected complications. Oracle recommends using the **Metal**, **Oracle**, or **CD/Motif** look and feel instead. Refer to "Working with TopLink Workbench Preferences" in *Oracle TopLink Developer's Guide* for complete information.

7.1.4 Unit of Work and JTA Transactions

If a unit of work is synchronized with a Java Transaction API (JTA), an exception will be thrown if it is released. If the current transaction requires that changes not be persisted, the JTA transaction must be rolled back. If you wish to have the transaction rolled back, do so with its API.

7.1.5 Using JDBC/ODBC Driver

Attempt to use the JDBC/ODBC driver throws the following exception:

```
Exception [TOPLINK-9002] (Oracle TopLink - 10g Release 3 (10.1.3.0.0) (Build
060118)): oracle.toplink.exceptions.SessionLoaderException [java]
Exception Description: Unable to load Project class
[oracle.toplink.demos.insurance.relational.InsuranceProject] [java]
Internal Exception: java.lang.NoClassDefFoundError: sun.jdbc.odbc.JdbcOdbcDriver
```

This exception occurs because HP JDK does not include JDBC-ODBC driver. Install a JDK that includes JDBC-ODBC driver of any other vendor to use JDBC-ODBC.

8

Oracle Business Rules

This chapter describes issues associated with Oracle Business Rules. It includes the following topics:

8.1 Using RL Reserved Words in Java Package Names

Invalid RL Language is generated if an RL Language reserved word (for example, the word `rule` in `mypkg.rule.com`) is part of the Java package name. If an RL Language reserved word is used in a Java package name, an error message like the following appears:

```
Oracle RL 1.0: syntax error PareseException: encountered 'rule' when expecting
one of: <XML_IDENTIFIER> ...<IDENTIFIER> ... "*" at line 11 column 19 in main
```

There is no workaround for this issue; do not use RL Language reserved words in Java package names.

8.2 Ancestor Methods are not Visible from Sub-Classes

The properties of a superclass are visible in the appropriate choice lists, but the methods of the ancestor classes are not visible.

There is no workaround for this issue.

8.3 New and Deleted Patterns are not Immediately Available

When you use Rule Author to add a pattern to a rule, the list of actions does not immediately reflect the existence of the new pattern instance. Similarly, when you use Rule Author to delete a pattern from a rule, the list of actions does not immediately reflect this change.

To work around this issue, you can do either of the following:

1. Update the rule by clicking either **OK** or **Apply**, then re-select the rule from the navigation pane.

2. Create a new action.

8.4 Changes to the Bind Variable Name are not Immediately Reflected

If you change the bind variable name on the Pattern Definition page, this change is not immediately reflected on the parent page (the Rule Author Rule page). In order to see your changes, you must close the Pattern Definition page before continuing your work on the Rule Author Rule page.

There is no workaround for this issue.

8.5 Setting the Location of the Oracle Wallet for Standalone OC4J

By default, Rule Author sets the location of the Oracle Wallet for use in password-protected WebDAV access to the following:

```
applications/ruleauthor/ruleauthor/WEB-INF/owd
```

When using OPMN in a full Oracle Application Server installation, this works correctly. When using a standalone OC4J instance, however, this path depends on the directory from which the standalone OC4J instance is started:

- If OC4J is started from *$ORACLE_HOME* by running `bin/oc4j -start`, the default path should be changed to:

  ```
  j2ee/home/applications/ruleauthor/ruleauthor/WEB-INF/owd
  ```

- If OC4J is started from *$ORACLE_HOME*/j2ee/home by running `java -jar oc4j.jar`, then the default path works correctly.

9

OracleAS Disaster Recovery

This chapter describes issues associated with OracleAS Disaster Recovery. It includes the following topics:

- Section 9.1, "General Issues and Workarounds"

9.1 General Issues and Workarounds

This section describes general issues and workarounds. It includes the following topic:

- Section 9.1.1, "Adding an Instance from a Remote Client Adds an Instance on the Local Instance and Not on the Remote Instance"

- Section 9.1.2, "Switchover Operation in an Asymmetric Topology Requires All Components to be Shutdown on Instances on the Primary Site that Do Not Have a Standby Peer"

- Section 9.1.3, "Error "prefix too long" Occurs During Image Backup"

9.1.1 Adding an Instance from a Remote Client Adds an Instance on the Local Instance and Not on the Remote Instance

When using the asgctl `add instance` command, the OracleAS Guard client must be run from a system that is already included in the topology.

For example, when an OracleAS Guard client is connected to the OracleAS Guard server that is to be added to an existing topology, the following error is returned:

```
ASG_IAS-15785: ERROR: The topology is missing the instance that exists in the home
where the ASG server is running.
You must first discover or add the instance in home
```

The workaround to this problem is to use an OracleAS Guard client from a system that is already included in the topology to perform the asgctl `add instance` command to add an instance to the topology.

9.1.2 Switchover Operation in an Asymmetric Topology Requires All Components to be Shutdown on Instances on the Primary Site that Do Not Have a Standby Peer

Prior to performing an asgctl switchover operation in an asymmetric topology for instances that do not have a standby peer, you must perform an opmnctl stopall command to shutdown all components on each of these ignored instances on the primary site.

When an XML policy file is in use for an asymmetric topology and has the <instanceList successRequirement ="Ignore" set for an instance, for example, as shown

in the following example, then in a switchover operation OracleAS Guard ignores that instance:

```
.
.
.
<instanceList successRequirement = "Ignore">
  <instance>instance B</instance>
</instanceList>
.
.
.
```

OracleAS Guard, on a switchover operation, shuts down all components on the old primary site except for OracleAS Guard and OPMN and ignores instance B because the policy file specifies to do so. The switchover operation fails because all components are not shut down on the primary site, in this case instance B because the policy file specifies to ignore instance B on the primary site, which has no standby peer.

To workaround this problem, the OracleAS Disaster Recovery Administrator must perform an opmnctl stopall operation for all components on instance B prior to the switchover operation in order for the switchover operation to succeed in this asymmetric topology.

9.1.3 Error "prefix too long" Occurs During Image Backup

The following command and option is used to create an image backup during the set of Oracle Application Server backup and recovery operations:

```
bkp_restore.sh -m node_backup -o image_backup -P
```

This command uses the first `tar` executable found in your `$PATH`. The command may fail with "prefix too long" errors due to long pathname limitations in the `/usr/bin/tar` executable that is installed by default on HP-UX PARISC systems. The workaround for this issue is to install a `tar` that does not have long pathname limitations into `/usr/local/bin` and place `/usr/local/bin` within `$PATH` before `/usr/bin` and before the Image Backup command is executed.

10

Oracle Sensor Edge Server

This chapter describes issues with Oracle Sensor Edge Server. It includes the following topics:

- Section 10.1, "Installation and Configuration Issues"
- Section 10.2, "General Issues"
- Section 10.3, "Documentation Issues"

10.1 Installation and Configuration Issues

This section describes installation and configuration issues regarding Oracle Sensor Edge Server. It includes the following topics:

- Section 10.1.1, "Valid 10.1.3 OC4J in Correct Oracle Home Required for Oracle Sensor Edge Mobile"
- Section 10.1.2, "OC4J 10.1.2 Must be Stopped if Installing Against It"
- Section 10.1.3, "Default Database Tablespaces"
- Section 10.1.4, "Oracle Sensor Edge Server Installation Fails"
- Section 10.1.5, "Possible Error When Installing SDR and SDS Simultaneously"

10.1.1 Valid 10.1.3 OC4J in Correct Oracle Home Required for Oracle Sensor Edge Mobile

You must have OC4J 10.1.3 installed into the same Oracle home into which you are going to install Oracle Sensor Edge Mobile. Having OC4J in another location is not supported by Oracle Universal Installer.

After confirming that OC4J is installed in the proper place, you can install Oracle Sensor Edge Server products. To install any Oracle Sensor Edge Server products, you must first select Oracle Sensor Edge Server in the Installation screen. Subsequently, you will be able to install Oracle Sensor Edge Server, Oracle Sensor Edge Mobile, or both products.

If you install both products, you will be given the option of launching the Sensor Edge Welcome twice. Both products are trying to open the same Welcome page; opening either one opens the same page.

10.1.2 OC4J 10.1.2 Must be Stopped if Installing Against It

As noted in Oracle Sensor Edge Server Guide, you can install Oracle Sensor Edge Server along with OC4J 10.1.2 (versus the current 10.1.3 version). If you choose to

install Oracle Sensor Edge Server with OC4J 10.1.2, ensure that OC4J is not running at the time. If OC4J 10.1.2 is running when you try to install Oracle Sensor Edge Server against it, the installation will fail.

10.1.3 Default Database Tablespaces

The scripts included with Oracle Sensor Edge Server are for a simple, generic installations. Our testing is accomplished against such generic installations.

However, in enterprise environments, there may be many custom requirements and configurations that customers must design and implement themselves.

For this reason, please ensure that you are familiar with tablespace creation. To learn about database tablespace creation, see the *Oracle SQL Reference* and *Oracle Database Concepts* guides for your particular release.

Once you have set up your database (including the server, datafiles, and tablespaces across datafiles), you can modify Oracle Sensor Edge Server install scripts as needed.

`create_edg_user.sql` is used for SDS, and `create_edg_sda_user.sql` is used for SDR.

These scripts must be modified to use newly-created tablespaces for specific customer requirements.

Under the create user command where you specify which tablespace the user is to use, change the line containing 'create user' in the .sql scripts, according to the manual.

10.1.4 Oracle Sensor Edge Server Installation Fails

When installing the Oracle Sensor Edge Server, do not select the Oracle Containers for J2EE 10.1.3.0.0 option from the companion CD Oracle Universal Installer (OUI); selecting this option along with the SES option causes the Oracle Sensor Edge Server installation to fail. The valid OC4J instance is installed using the Oracle Application Server CD-ROM (or DVD-ROM), not the companion CD OUI. For more information, see the Oracle Application Server Installation Guide appropriate to your platform.

10.1.5 Possible Error When Installing SDR and SDS Simultaneously

If you attempt to install both the SDR and SDS options during the same installation (that is, both options checked in the OUI), then you must ensure that the SYS and SYSTEM passwords on the target database are the same. If the passwords are different, the installation will fail.

10.2 General Issues

This section describes general issues encountered in Oracle Sensor Edge Server. It includes the following topics:

- Section 10.2.1, "Using UTL_EDG.REMOVE_RULE Displays an Error"
- Section 10.2.2, "Adding a Rule Displays an Error"
- Section 10.2.2, "Adding a Rule Displays an Error"
- Section 10.2.3, "Localization -- Navigation Tree in the SES Console Renders as the Server Locale-Defined Character Set"
- Section 10.2.4, "Reassignment of Audio Event Type (207)"

10.2.1 Using UTL_EDG.REMOVE_RULE Displays an Error

When using the `API UTL_EDG.REMOVE_RULE`, an error may be encountered. This API only works if your rule was created through other Oracle Sensor Edge Server APIs.

10.2.2 Adding a Rule Displays an Error

In this release, be sure to use the syntax `':event'` instead of `'TAB.USER_DATA'`.

10.2.3 Localization -- Navigation Tree in the SES Console Renders as the Server Locale-Defined Character Set

The character set defined in the Server Locale overrides the characters set in the browser locale even when you switch the browser locale. For example, if you switch the browser locale to *Japanese*, the Server Locale is defined as traditional Chinese, the strings in the Navigation Tree of the SES Console (such as the *Available Extensions*, filters, devices, and device group nodes) and the error and confirmation messages render in traditional Chinese rather than in Japanese. To correct this problem, reset the Server Locale, or override the Server Locale-defined character set by overloading *MsgTranslator* methods to take locale. String literals in the SESConfig tree control must be non-static and fetched using *Locale* on each page request.

10.2.4 Reassignment of Audio Event Type (207)

The Audio Event Type (which is the event supported by the Simple Audio Driver) has been reassigned from *207* to *104*, and is now included in the group Generic Instructions to Devices (Event Types 100 - 199). Its Subtype remains *1* (*Play audio jobs in the .xml file in the Datafield*).

10.3 Documentation Issues

This section describes documentation issues regarding Oracle Sensor Edge Server. It includes the following topic:

- Section 10.3.1, "Documentation for Oracle Sensor Edge Server Extensions"
- Section 10.3.2, "Manually Deploying Sensor Data Streams Against an Existing Sensor Data Repository"

10.3.1 Documentation for Oracle Sensor Edge Server Extensions

Once the server is installed, you can see documentation for extensions at: *http://<host on which SES is installed>:<oc4j port>/edge/extensions*. Or, see Oracle Sensor Edge Server information on Oracle Technology Network at: `http://www.oracle.com/technology/products/sensor_edge_server/extensions.html`.

10.3.2 Manually Deploying Sensor Data Streams Against an Existing Sensor Data Repository

Oracle Sensor Edge Server Guide omits a step in describing how to manually deploy Sensor Data Streams if you have already created a Sensor Data Repository. The procedure for manually deploying Sensor Data Streams if the Sensor Data Repository exists is as follows:

1. Use SQL*Plus to connect to the database as *sysdba*, by running `sqlplus /nolog`.

2. Run sqlplus; connect as *sys/your_pwd@your_db as sysdba*.

3. Run the script `grant_edg_user.sql`, located in `Oracle_Home /edge/stage/sql/10.1.3`.

4. Disconnect as *sys* and then reconnect to the database as the *edge user*.

5. Run the script `edg_create_streams.sql`, located in `Oracle_Home /edge/stage/sql/10.1.3`.

6. Follow the procedure described in "Connecting Oracle Sensor Edge Server to Sensor Data Streams".