# Oracle® Real-Time Decisions

Release Notes

Versions 2.2, 2.2.1, and 2.2.1.1

**E10049-06**

November 2009

Oracle Real-Time Decisions (Oracle RTD) enables you to develop adaptive enterprise software solutions. These adaptive solutions continuously learn from business process transactions while they execute and optimize each transaction, in real time, by way of rules and predictive models.

These release notes describe known issues and workarounds for Oracle RTD Versions 2.2, 2.2.1, and 2.2.1.1.

For information about installing Oracle RTD, consult *Oracle Real-Time Decisions Installation and Administration of Oracle RTD*, which applies to Versions 2.2, 2.2.1, and 2.2.1.1.

For information about new features which apply to Versions 2.2.1 and 2.2.1.1, consult *Oracle Real-Time Decisions New Features Guide*.

Both *Oracle Real-Time Decisions Installation and Administration of Oracle RTD* and *Oracle Real-Time Decisions New Features Guide* are available on the Oracle RTD DVD ROM and from the Oracle RTD Documentation Web site:

http://www.oracle.com/technology/documentation/rtd.html

The following list describes the sections of this document:

- Section 1, "How to Use These Release Notes"
- Section 2, "Upgrade Issues"
- Section 3, "General Issues and Workarounds"
- Section 4, "Configuration Issues"
- Section 5, "Administration Issues"
- Section 6, "Integration Issues"
- Section 7, "Oracle RTD Decision Studio Issues"
- Section 8, "Oracle RTD Decision Center Issues"
- Section 9, "Load Generator Issues"
- Section 10, "Globalization Issues"
- Section 11, "Documentation Errata"
- Section 12, "Documentation Accessibility"

**ORACLE**®

# 1 How to Use These Release Notes

These release notes are updated periodically as new information becomes available. To ensure that you are reading the latest version of the release notes, check the Oracle RTD Documentation Web site:

http://www.oracle.com/technology/documentation/rtd.html

# 2 Upgrade Issues

This section describes issues associated with upgrading to Oracle RTD Versions 2.2, 2.2.1, and 2.2.1.1 from previous versions, and consists of the following topics:

- Section 2.1, "Upgrading to Oracle RTD Version 2.2 from Previous Releases"
- Section 2.2, "Upgrading from Oracle RTD Version 2.2 to Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1"

## 2.1 Upgrading to Oracle RTD Version 2.2 from Previous Releases

This section consists of the following topic:

- Section 2.1.1, "Must Change All User Passwords After Upgrading to Oracle RTD"

### 2.1.1 Must Change All User Passwords After Upgrading to Oracle RTD

Oracle RTD Version 2.2 uses a new, more secure Message Digest Algorithm (SHA-1). If you are upgrading to Oracle RTD Version 2.2 from a previous release, and you are using Oracle RTD Platform Authentication, you should re-enter or change all existing user passwords. This action is necessary because the old passwords were set under a different Message Digest Algorithm and will not work.

Alternatively, you can revert the Message Digest Algorithm to the previous version (MD5). To do this, use JConsole to update the MessageDigestAlgorithm property of the OracleRTD > SDClusterPropertyManager > Security Properties MBean. See *Oracle Real-Time Decisions Installation and Administration of Oracle RTD* for more information about using JConsole with Oracle RTD.

Note that the MD5 Message Digest Algorithm uses a weaker encryption and is more vulnerable to security violations.

## 2.2 Upgrading from Oracle RTD Version 2.2 to Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1

This section describes how to upgrade from Oracle RTD Version 2.2 to Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1.

The assumptions are:

- Oracle RTD Version 2.2 was installed at C:\OracleBI\RTD.
- The Oracle RTD Version 2.2 database was initialized.
- Your Inline Services were deployed on Oracle RTD Version 2.2.
- The Oracle RTD Version 2.2 Decision Studio workspace is at the default location.
  - On Windows, the default location is *%USERPROFILE%*\Oracle RTD Studio, for example, C:\Documents and Settings\*your user name*\Oracle RTD Studio.

This section consists of the following topics, each of which represents a task to perform during the upgrade process:

- Section 2.2.1, "Shutting Down Oracle RTD Version 2.2"

- Section 2.2.2, "Performing a Backup of Existing Data"

- Section 2.2.3, "Unpacking Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1"

- Section 2.2.4, "Upgrading the Oracle RTD Decision Studio Workplace"

- Section 2.2.5, "Upgrading the Oracle RTD Database"

- Section 2.2.6, "Deploying Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1"

### 2.2.1  Shutting Down Oracle RTD Version 2.2

To shut down Oracle RTD Version 2.2:

1. Exit from Oracle RTD Decision Studio.

2. Exit from Oracle RTD Load Generator.

3. Open your J2EE application server's administration console.

4. Stop RTD.

5. Undeploy RTD.

6. Shut down your J2EE application server.

### 2.2.2  Performing a Backup of Existing Data

To back up your existing data:

1. Rename `C:\OracleBI\RTD` to `C:\OracleBI\RTD.22`.

2. Copy `C:\Documents and Settings\`*your user name*`\Oracle RTD Studio` to `C:\Documents and Settings\`*your user name*`\Oracle RTD Studio.22`.

3. Back up your Oracle RTD database.

   Refer to your database vendor's documentation for information about backing up a database.

### 2.2.3  Unpacking Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1

To unpack Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1:

1. Expand the Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1 distribution archive to the desired location, for example, `C:\OracleBI\RTD`.

### 2.2.4  Upgrading the Oracle RTD Decision Studio Workplace

To upgrade the Oracle RTD Decision Studio Workplace:

1. Start Oracle RTD Decision Studio.

2. If you have any compile errors, select **Project > Clean > Clean all projects**.

   If you still have compile errors, there may also be problems with the Oracle RTD server. For help resolving these issues, contact your local Oracle support organization.

### 2.2.5  Upgrading the Oracle RTD Database

To upgrade your Oracle RTD database:

1. Start SDDBTool.

   For information about SDDBTool, see *Oracle Real-Time Decisions Installation and Administration of Oracle RTD*.

2. When the option to Initialize or Upgrade appears, select **Upgrade**.

### 2.2.6  Deploying Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1

To deploy Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1:

1. Start your J2EE application server.

   For information about installing the RTD.ear file, see *Oracle Real-Time Decisions Installation and Administration of Oracle RTD*.

2. In the RTD server.log file, check that your existing Inline Services loaded successfully, by examining the [AppFactory] entries.

   For example,

   ```
   2007-11-26 17:42:46,725 INFO [AppFactory] Loaded Inline
   Service DC_Demo:4 in deployment state Development.
   ```

3. Alternatively, you can check the status of your Inline Services through JMX. In JConsole, navigate to **MBeans > Oracle RTD > Inline Service Manager >** *Inline Service Name*.
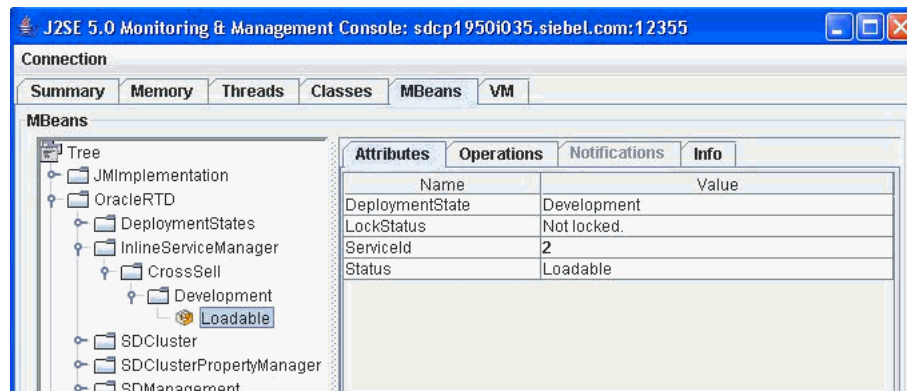


   Expand the *Inline Service Name* to show the deployment state of interest, and note the Status of the deployment state. If the Status attribute has a value of **Loadable**, then the Inline Service for that deployment state has been loaded properly and can now accept requests. If the server was unable to upgrade and load your existing Inline Service, the Status attribute will show a value of **Failed**.

   In this case, you can deploy a working version of the Inline Service from the Decision Studio. If you no longer have the source code for an Inline Service, you can download the Inline Service, using **Project > Download**.

## 3  General Issues and Workarounds

This section describes general issues and workarounds for Oracle RTD. It contains the following topics:

- Section 3.1, "Certification Information"

- Section 3.2, "Installation Issues"

## 3.1  Certification Information

For the latest certification information, which supersedes that in the Oracle RTD documentation, refer to *System Requirements and Supported Platforms for Oracle Real-Time Decisions* for Versions 2.2, 2.2.1, and 2.2.1.1, available on the Oracle RTD Documentation Web site at:

http://www.oracle.com/technology/documentation/rtd.html

## 3.2  Installation Issues

This section provides release notes on installing Oracle RTD. It contains the following topics:

### 3.2.1  Must Not Install Version 1.6 of Sun JDK

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

For each of the application servers OC4J, WebSphere, and WebLogic, when you must install a Java Development Kit, ensure that you install Sun JDK Version 1.5.X or above, *but not Version 1.6*.

See *Oracle Real-Time Decisions Installation and Administration of Oracle RTD* for more information about installing the Java Development Kit.

### 3.2.2  Must Unzip Oracle RTD Files In a Location That Does Not Include Spaces

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

When you install Oracle RTD, you must unzip Oracle RTD in a directory location that does not include any spaces in the path. Otherwise, there will be server-side problems, and Load Generator logging will not work.

### 3.2.3  Deploying Oracle RTD in OC4J on IBM ThinkPad Laptops Causes Blue Screen Error

This issue applies to Version 2.2 only.

When you deploy Oracle RTD in OC4J on an IBM ThinkPad laptop, you may encounter a blue screen error due to an IBM issue.

To avoid this problem, upgrade the ThinkVantage Rescue and Recovery software for the laptop to version 3.x.

### 3.2.4  Deploying Oracle RTD in OC4J Into a Container Not Named "Home"

This issue applies to Version 2.2 only. It is fixed in Version 2.2.1 and Version 2.2.1.1.

To deploy Oracle RTD in OC4J into a container that is not "home", you must edit the file `orion-application.xml` in the `RTD.ear` file before deploying Oracle RTD into the container.

The `RTD.ear` file is in the *RTD_HOME*\package directory.

Perform the following steps:

**1.** Create the new OC4J instance, with the new name.

**2.** In the *RTD_HOME*\package directory, locate the `RTD.ear` file.

**3.** Make a copy of RTD.ear, and store it outside of the *RTD_HOME*\package directory, or rename it with a different extension, for example, `RTD.ear.orig`.

**4.** Open the `RTD.ear` file in *RTD_HOME*\package with Winzip or another archival tool, such as Winrar.

**5.** Extract the file orion-application.xml.

   Ensure that you check **Use folder names**, so that the file `orion-application.xml` goes into the *RTD_HOME*\package\META-INF directory
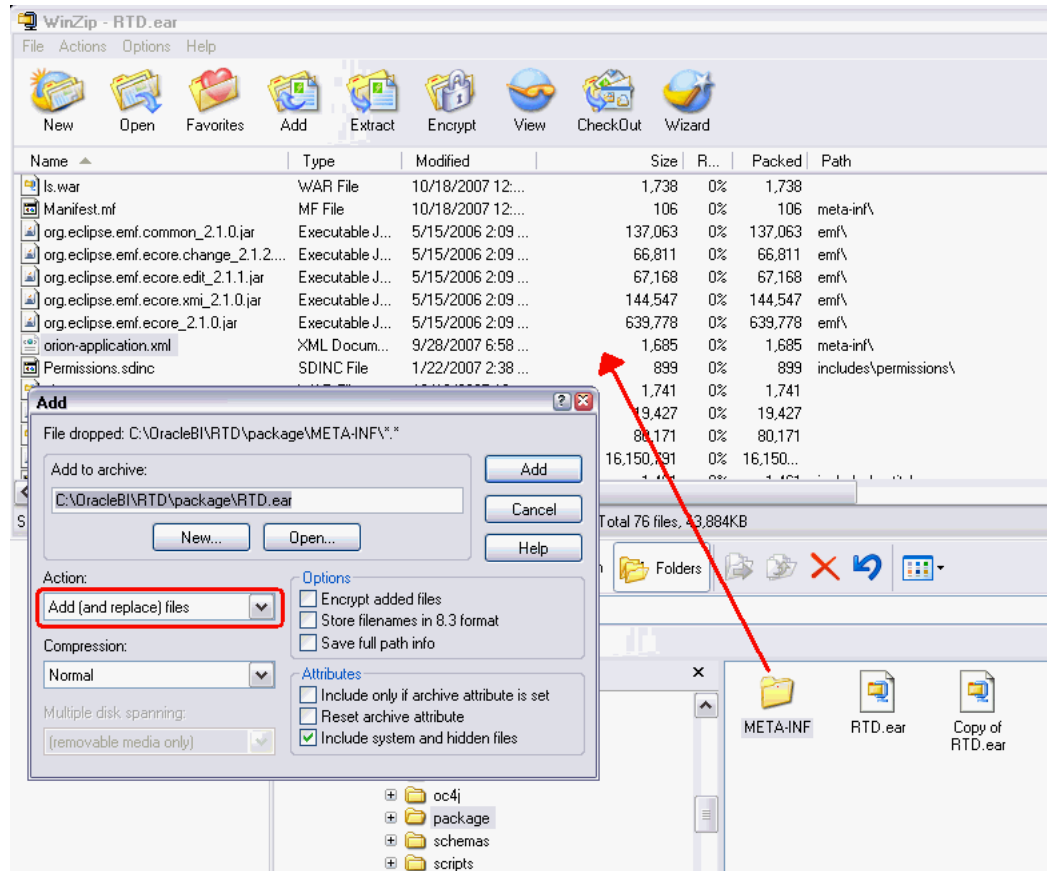
**6.** Edit the file `orion-application.xml` in the `META-INF` directory, by changing the following lines:

```
<library path="${oracle.home}/j2ee/home/applications/OracleRTD/ui-libs/" />
<library path="${oracle.home}/j2ee/home/applications/OracleRTD/sd.jar/" />
<library path="${oracle.home}/j2ee/home/applications/OracleRTD/deploy/" />
```

   to:

```
<library path="${oracle.j2ee.home}/applications/OracleRTD/ui-libs/" />
<library path="${oracle.j2ee.home}/applications/OracleRTD/sd.jar" />
<library path="${oracle.j2ee.home}/applications/OracleRTD/deploy/" />
```

**7.** With the META-INF folder visible in Windows explorer, as well as `RTD.ear` opened with Winzip, drag the META-INF folder into the Winzip window, as in the following screenshot:

8. In the Add window, ensure that Action is set to **Add (and replace) files**, and click **Add**.

9. Deploy RTD into the new container.

### 3.2.5 Coexistence of Oracle RTD and Other Applications in OC4J Containers

This issue applies to Version 2.2 only. It is fixed in Version 2.2.1 and Version 2.2.1.1.

When Oracle RTD and other applications are in the same OC4J container, there are XML parser problems. The workaround is to deploy Oracle RTD in a separate OC4J container.

For more information on deploying Oracle RTD in a separate OC4J container, see Section 3.2.4, "Deploying Oracle RTD in OC4J Into a Container Not Named "Home"."

### 3.2.6 Oracle RTD Startup Problem Creating Sockets, Caused By IP_ADD_ MEMBERSHIP Error

This issue applies to Versions 2.2.1 and 2.2.1.1 only.

This error occurs during Oracle RTD startup on some machines running various versions of Microsoft Windows, when the DHCP Media Sense feature is enabled, as it is by default. The error prevents Oracle RTD from starting up.

The error is characterized by the following messages in close proximity in the server.log file:

- java.lang.Exception: problem creating sockets

- java.net.SocketException: IP_ADD_MEMBERSHIP failed (out of hardware filters?)

To avoid this problem, you must make the Microsoft Windows registry modification described in the Microsoft KnowledgeBase article, *How to Disable the Media Sensing Feature for TCP/IP in Windows*, at:

`http://support.microsoft.com/kb/239924.`

## 3.3 General Issues

This section contains the following topics:

- Section 3.3.1, "Expanding Model Inputs Causes Null Pointer Exception"
- Section 3.3.2, "Changing Client Examples if Oracle RTD Server Port is Not 8080"
- Section 3.3.3, "Siebel Object Manager Authentication Not Supported in Oracle RTD Version 2.2.1 and Version 2.2.1.1"
- Section 3.3.4, "Alerts Removed From Oracle RTD Version 2.2.1 and Oracle RTD Version 2.2.1.1"
- Section 3.3.5, "Choice Group Revenue Report Removed From Oracle RTD Version 2.2.1 and Oracle RTD Version 2.2.1.1"
- Section 3.3.6, "Model Snapshot Not Recording Parent Choice Group Name Of Dynamic Choices"
- Section 3.3.7, "StringCatalog Throws SQLExceptions From Not Passing Nulls"
- Section 3.3.8, "Infinite Request For Forwarding Timeouts"
- Section 3.3.9, "Entity Array Reporting is Broken"

### 3.3.1 Expanding Model Inputs Causes Null Pointer Exception

This issue applies to Version 2.2 only.

If you change or expand the model inputs after a model already has data in it (for example, if you add an attribute to a session-dependent entity), then the learningDispatcher throws a null pointer exception, and all new learning records are not processed.

If you are in the development phase, you can work around this problem by using JConsole to remove all operational data, as follows:

1. If you are using OC4J or WebLogic, open JConsole by running *JAVA_ HOME*\bin\jconsole.exe. If you are using WebSphere, run the batch script you created during JConsole configuration. See *Oracle Real-Time Decisions Installation and Administration of Oracle RTD* for more information about accessing JConsole.

2. Click the **Remote** tab. Then, enter the appropriate port number (typically 12345) and the administrator credentials you created during installation and click **Connect**.

3. Click the **MBean** tab, then go to the **OracleRTD > InlineServiceManager > *Inline_ Service_Name > Deployment_State* > Loadable** MBean.

4. Click the **Operations** tab, then use the **deleteAllOperationalData()** operation to remove all operational data, including the study, for this Inline Service.

If you encounter this problem in a production deployment of Oracle RTD, contact Oracle Support Services.

### 3.3.2  Changing Client Examples if Oracle RTD Server Port is Not 8080

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

If the Oracle RTD Server port is not 8080, you must edit the client properties information for the integration method you have selected for Oracle RTD. The client example files exist in the directory *RTD_HOME*/client/Client Examples.

From the sections that follow, select and perform the client example appropriate to your selected integration method.

**Dot Net Client Example**

The Dot Net Client example is a Microsoft Visual Studio C# client program.

To change the server port:

1.  In Visual Studio, select **File > Open > Project...**

2.  For File Name, select *RTD_HOME*\client\Client Examples\Dot NET Client Example\DotNetSmartClientExample.sln, and click **Open**.

3.  Open *RTD_HOME*\client\Client Examples\Dot NET Client Example\DotNetSmartClientExample.cs, by double-clicking **DotNetSmartClientExample.cs** in the Solution Explorer window on the right.

4.  Search for the entry:

    ```
    SDClient client = new SDClient("http://localhost:8080");
    ```

5.  Change the URL localhost:8080 to match the host and port of the Oracle RTD server that you are using.

6.  Save the file.

7.  Click **Debug > Start**.

**Java Smart Client Example**

To change the server port:

1.  In Eclipse, select **File > Import**.

2.  Select **Existing Projects into Workspace**, and click **Next**.

3.  Select **Select root directory**, and click **Browse**.

4.  Select the directory RTD_HOME\client\Client Examples\Java Client Example, and click **Ok**.

5.  On the left, expand the **Java Smart Client Example** tree, and open the **lib** directory.

6.  Open or double-click the file sdclient.properties.

7.  Search for the entry:

    ```
    HTTP1.url = http://localhost:8080
    ```

8.  Change the URL localhost:8080 to match the host and port of the Oracle RTD server that you are using.

9.  Save the file.

10. Switch to the Java perspective.

11. Click **Run > Run.**

12. Click **Java Application** in the configurations list on the left.

13. Click **New** at the bottom left.

14. Enter a Name, for example, `Java Smart Client Example`.

15. In the Main tab:

    1. Click **Browse**, and select **JavaSmartClientExample**.

    2. Click **Search**.

    3. For the Example class, select **com.sigmadynamics.client.example.Example**.

16. Click **Apply**, and save the configuration when prompted.

17. Click **Run**.

### JSP Client Example

To change the server port:

1. Open *RTD_HOME*`\client\Client Examples\JSP Client Example\sdclient-test.war` with WinZip or WinRAR.

2. In sdclient-test.war, open `client\sdclient.properties`.

3. Search for the entry:

   ```
   HTTP1.url = http://localhost:8080
   ```

4. Change the URL `localhost:8080` to match the host and port of the Oracle RTD server that you are using.

5. Save the file back into `sdclient-test.war`.

6. Deploy the war application in either OC4J, WebLogic, or WebSphere.

   For more information, see *Oracle Real-Time Decisions Installation and Administration of Oracle RTD*.

### 3.3.3 Siebel Object Manager Authentication Not Supported in Oracle RTD Version 2.2.1 and Version 2.2.1.1

This issue applies to Versions 2.2.1 and 2.2.1.1 only.

Siebel Object Manager Authentication is not a supported authentication method in Versions 2.2.1 and 2.2.1.1.

All references to Siebel Object Manager Authentication in *Oracle Real-Time Decisions Installation and Administration of Oracle RTD* are no longer valid.

These references appear mainly in Chapter 6, Configuring Authentication for Oracle Real-Time Decisions, and particularly in Section 6.4, Configuring Oracle's Siebel Object Manager Authentication.

In addition, in Chapter 10, JMX Management of Oracle Real-Time Decisions, in Section 10.5.2, About OracleRTD > SDClusterPropertyManager > Security Properties, the value `.com.sigmadynamics.server.security.SiebelAuthenticator` is no longer an option for the attribute **AuthenticationProviderClass**.

### 3.3.4 Alerts Removed From Oracle RTD Version 2.2.1 and Oracle RTD Version 2.2.1.1

This issue applies to Versions 2.2.1 and 2.2.1.1 only.

Alerts are not supported in Oracle RTD Versions 2.2.1 and 2.2.1.1.

### 3.3.5  Choice Group Revenue Report Removed From Oracle RTD Version 2.2.1 and Oracle RTD Version 2.2.1.1

This issue applies to Versions 2.2.1 and 2.2.1.1 only.

The choice group Revenue report, previously available as a Performance report in the Decision Center, has been removed from Oracle RTD Version 2.2.1 and Oracle Version 2.2.1.1.

### 3.3.6  Model Snapshot Not Recording Parent Choice Group Name Of Dynamic Choices

This issue applies to Versions 2.2 and 2.2.1 only.

When model snapshots of dynamic choices are triggered, the parent Choice Group names are not being recorded in the RTDChoiceGroup table, which leads to  all dynamic choices ending up as children of the BASE_EVENT Choice Group.

### 3.3.7  StringCatalog Throws SQLExceptions From Not Passing Nulls

This issue applies to Versions 2.2 and 2.2.1 only.

StringCatalog throws a SQLException, because StringCatalog is not initializing all the parameters being passed to its stored procedure, assuming they would be initialized automatically to null, but they sometimes have values from a previous call.

### 3.3.8  Infinite Request For Forwarding Timeouts

This issue applies to Versions 2.2 and 2.2.1 only.

When Oracle RTD forwards an asynchronous request from one server to another, it uses no timeout. The thread that issued the forwarding request may fail to finish in certain situations, for example, if the message became lost because of a network failure. This would consume memory and other resources, eventually leading to poor performance and even to the need to restart the server.

An asynchronous request is a Decision Service Integration Point request that returns no values; it is also known as an Informant. Forwarding would occur in a clustered installation of Oracle RTD Decision Service instances if a request arrives at a specific Oracle RTD Decision Service instance, when the request's session key is that of a session residing on a different Oracle RTD instance.

### 3.3.9  Entity Array Reporting is Broken

This issue applies to Versions 2.2, 2.2.1, and 2.2.1.1.

Decision Center entity reports do not show any values for attributes of an entity that are entities themselves. For instance, if we have a Customer entity, and within that Customer entity we have an Assets attribute of type Assets entity, none of the attributes tied to the Assets entity will be visible in the reports for the Customer entity, nor will they be visible in the reports for the Assets entity.

## 4  Configuration Issues

This section provides release notes related to Oracle RTD configuration. It contains the following topics:

- Section 4.1, "Oracle BI Enterprise Edition 10.1.3.2 and Oracle's Siebel Analytics Server Not Supported as Enterprise Data Sources when Oracle RTD is Deployed on WebSphere"

## 4.1 Oracle BI Enterprise Edition 10.1.3.2 and Oracle's Siebel Analytics Server Not Supported as Enterprise Data Sources when Oracle RTD is Deployed on WebSphere

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

When Oracle RTD is deployed on WebSphere, Oracle BI Enterprise Edition (Oracle BI EE) and Oracle's Siebel Analytics Server are not supported as data sources for Oracle RTD.

## 4.2 Decision Service Only Accepts Requests from the Local Host by Default

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

---

**Note:** In Versions 2.2.1 and 2.2.1.1, the final step of the process described in this section (Restart Oracle RTD) is not required.

---

In previous releases, Oracle RTD Decision Service requests were not protected. In Oracle RTD Version 2.2, for greater security, the Decision Service by default will only accept requests from the computer that hosts the Decision Service.

For production environments where Real-Time Decision Server is installed on a separate host, administrators can set up a list of trusted hosts from which the Decision Service can receive requests.

---

**Note:** Remote Decision Center access (browser access) is not affected by this feature.

---

To specify a list of trusted hosts for Decision Service requests:

1. If you are using OC4J or WebLogic, open JConsole by running *JAVA_HOME*/bin/jconsole.exe. If you are using WebSphere, run the batch script you created during JConsole configuration. See *Oracle Real-Time Decisions Installation and Administration of Oracle RTD* for more information about accessing JConsole.

2. Click the **Remote** tab. Then, enter the appropriate port number (typically 12345) and the administrator credentials you created during installation and click **Connect**.

3. Click the **MBean** tab, then go to the **OracleRTD > SDClusterPropertyManager > Cluster** MBean and ensure the **RestrictDSClients** attribute is set to `true`. This attribute ensures that the Decision Service only accepts requests from its own host, or from the list of hosts identified in the **TrustedDSClients** attribute.

4. In the **OracleRTD > SDClusterPropertyManager > Cluster** MBean, update the **TrustedDSClients** attribute to include a semicolon-separated list of IP addresses of the hosts from which you want Decision Service to accept requests. You must specify IP addresses; do not specify host names.

5. *For Version 2.2 only:* Restart Oracle RTD.

## 4.3  Communication Between Multiple Real-Time Decision Servers Is Restricted by Default

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

> **Note:**   In Versions 2.2.1 and 2.2.1.1, the final step of the process described in this section (Restart Oracle RTD) is not required.

In previous releases, internal communication between multiple Real-Time Decision Servers was not protected. In Oracle RTD Version 2.2, for greater security, multiple Real-Time Decision Servers cannot communicate with each other by default.

For production environments where multiple Real-Time Decision Servers are installed, administrators can set up a list of trusted cluster hosts so that the Real-Time Decision Servers can communicate with each other.

To specify a list of trusted cluster hosts for Real-Time Decision Server internal communication:

1. If you are using OC4J or WebLogic, open JConsole by running *JAVA_HOME*/bin/jconsole.exe. If you are using WebSphere, run the batch script you created during JConsole configuration. See *Oracle Real-Time Decisions Installation and Administration of Oracle RTD* for more information about accessing JConsole.

2. Click the **Remote** tab. Then, enter the appropriate port number (typically `12345`) and the administrator credentials you created during installation and click **Connect**.

3. Click the **MBean** tab, then go to the **OracleRTD > SDClusterPropertyManager > Cluster** MBean and ensure the **RestrictClusterMembers** attribute is set to `true`. This attribute ensures that the Real-Time Decision Servers can only communicate with the hosts listed in the **TrustedClusterMembers** attribute.

4. In the **OracleRTD > SDClusterPropertyManager > Cluster** MBean, set the **SDGroupName** attribute to some non-blank value.

5. In the **OracleRTD > SDClusterPropertyManager > Cluster** MBean, update the **TrustedClusterMembers** attribute to include a semicolon-separated list of IP addresses of the hosts where the Real-Time Decision Servers are installed. You must specify IP addresses; do not specify host names.

6. *For Version 2.2. only:* Restart Oracle RTD.

## 4.4 Configuration Properties WebServerPort and HTTPSEnabled Are Not Used by Oracle RTD Version 2.2.1 nor Oracle RTD Version 2.2.1.1

This issue applies to Versions 2.2.1 and 2.2.1.1 only.

The JMX properties **WebServerPort** and **HTTPSEnabled**, which were previously used for internal communication between Oracle RTD components, are no longer used in Versions 2.2.1 and 2.2.1.1.

For more details on how this affects the configuration setup steps, see Section 11.11, "Configuration Properties WebServerPort and HTTPSEnabled Not Required in Version 2.2.1 Configuration Procedures."

## 4.5 Import Database Table Box Returns All Tables from All Catalogs for an Oracle BI EE Data Source

This issue applies to Versions 2.2.1 and 2.2.1.1 only.

When you enter the URL for an Oracle BI EE data source, you can specify a value for the parameter `catalog`. The intention of the `catalog` parameter is to enable you to view and select from just the tables in a particular catalog.

The error is that all tables in all catalogs appear in the Import Database Table window, even when you specify a value for the `catalog` parameter. This situation is undesirable because different catalogs may have tables of the same name.

In the process of creating a data source in Oracle RTD Decision Studio, if you select an incorrect table name, you can either delete the data source, or manually alter the column names of the data source.

## 4.6 JDBC DataSource Class Support For Oracle Database

This issue applies to Versions 2.2 and 2.2.1 only.

The Oracle RTD SDDS JDBC provider connection factory class `oracle.jdbc.pool.OracleDataSource` is now supported in Oracle RTD 2.2.1.1 for OracleAS. *Oracle Real-Time Decisions Installation and Administration of Oracle RTD* mentions the use of only `oracle.jdbc.driver.OracleDriver`. The **OracleDataSource** jdbc class provides better database failover.

# 5 Administration Issues

This section provides release notes on Oracle RTD administration. It contains the following topics:

- Section 5.1, "Cannot Create New Deployment States in JConsole"

- Section 5.2, "JGroup Error Appears in Log When Oracle RTD is Deployed on OC4J"

- Section 5.3, "Must Restart OC4J When Database Server Restarts"

- Section 5.4, "For JConsole on WebSphere, InlineServiceManager Does Not Refresh After Inline Services Are Deployed"

## 5.1 Cannot Create New Deployment States in JConsole

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

You cannot use JConsole to create new Deployment States, or modify existing deployment states. You can only use the three default Deployment States: Development, QA, and Production.

This issue has no workaround.

## 5.2  JGroup Error Appears in Log When Oracle RTD is Deployed on OC4J

This issue applies to Version 2.2 only.

When Oracle RTD is deployed on OC4J, a Jgroups exception appears in the Oracle RTD log when you stop Oracle RTD. This error is harmless and can be ignored.

## 5.3  Must Restart OC4J When Database Server Restarts

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

When Oracle RTD is running on standalone OC4J, if the Oracle RTD Database goes down for any reason, you must restart OC4J.

This problem occurs because standalone OC4J does not support refreshing stale database connections after a database restart.

## 5.4  For JConsole on WebSphere, InlineServiceManager Does Not Refresh After Inline Services Are Deployed

This issue applies to Version 2.2 only.

When Oracle RTD is deployed on WebSphere, the JConsole administration tool does not automatically refresh the InlineServiceManager MBean after an Inline Service is deployed to Real-Time Decision Server. Because of this problem, the InlineServiceManager MBean in JConsole may not have the latest list of deployed Inline Services.

To work around this problem in JConsole, first click the OracleRTD > SDManagement > InlineServiceManager MBean before accessing the OracleRTD > InlineServiceManager MBean. Clicking the OracleRTD > SDManagement > InlineServiceManager MBean refreshes the list of deployed Inline Services.

# 6  Integration Issues

This section provides release notes on Oracle RTD integration. It contains the following topics:

- Section 6.1, "sdclient.properties File Does Not Contain Property for Enabling Client-Side Timeouts by Default"
- Section 6.2, "NullPointerException Occurs When Client-Side Timeouts are Enabled for Java Smart Client"
- Section 6.3, "Java Smart Client TCP Connection Leak"

## 6.1  sdclient.properties File Does Not Contain Property for Enabling Client-Side Timeouts by Default

This issue applies to Version 2.2 only.

When integrating to Oracle RTD using the Java Smart Client, the property to enable client-side timeouts does not appear by default in the `sdclient.properties` file.

To enable client-side timeouts for the Java Smart Client, follow these steps:

1. Open the file *RTD_HOME*`\client\Client Examples\Java Client Example\lib\sdclient.properties` for editing.

2. Add the following line:

   ```
   clientTimeoutsEnabled=true
   ```

3. Save and close the file.

## 6.2 NullPointerException Occurs When Client-Side Timeouts are Enabled for Java Smart Client

This issue applies to Version 2.2 only.

If client-side timeouts are enabled for the Java Smart Client, a `NullPointerException` is thrown when you attempt to test the default offer functionality. In addition, the server becomes unreachable.

To work around this problem, follow these steps:

1. Open the file *RTD_HOME*`\client\Client Examples\Java Client Example\src\com\sigmadynamics\client\example\Example.java` for editing.

2. At the bottom of the file, add the following code to the beginning of the `if` statement in the method `checkIfConnectionLost()`:

   ```
   e.getCause() == null ||
   ```

   For example:

   ```
     public static void checkIfConnectionLost( SDClientException e, String
   integrationPointName, String message )throws SDClientException{
       if (e.getCause() == null || e.getCause() instanceof ConnectException ||
   e.getCause().getCause() instanceof ConnectException)
           System.out.println("Unable to contact server for '" +
   integrationPointName + "'. " + message);
       else
         throw e;
     }
   ```

3. Save and close the file.

4. Retest the default offer functionality of the Java Smart Client.

## 6.3 Java Smart Client TCP Connection Leak

This issue applies to Versions 2.2 and 2.2.1 only.

The Java Smart Client does not close its TCP connections, resulting in a connection leak. Even when the server closes the connection, the socket is left in a CLOSE_WAIT state for up to two minutes or until the client process terminates. Because the server has a limited number of sockets, typically several thousand, clients could experience socket-connect errors during periods of heavy traffic.

# 7 Oracle RTD Decision Studio Issues

This section provides release notes for Oracle RTD Decision Studio. It contains the following topic:

- Section 7.1, "Deploying an Inline Service from Decision Studio Causes a Null Pointer Exception in the WebLogic Log"

- Section 7.2, "Non-Default Decision Studio Workspace Path Not Persistent"

- Section 7.3, "Explicitly Defining Partitioning Attribute Values Causes Subsequent Decision Center Report Errors"

## 7.1 Deploying an Inline Service from Decision Studio Causes a Null Pointer Exception in the WebLogic Log

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

When Oracle RTD is running on WebLogic, when you deploy an Inline Service from Decision Studio, a null pointer exception appears in the WebLogic log. This error is harmless and can be ignored.

## 7.2 Non-Default Decision Studio Workspace Path Not Persistent

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

The default Decision Studio workspace path is `C:\Documents and Settings\`*`<user_name>`*`\Oracle RTD Studio`. Users can set the Decision Studio workspace path to a non-default location, but the next time that Decision Studio is started up, it uses the default again.

## 7.3 Explicitly Defining Partitioning Attribute Values Causes Subsequent Decision Center Report Errors

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

When an Oracle RTD model is defined with partitions, then at run-time, when each new partitioning attribute value is presented to the learning process, Oracle RTD automatically creates a new partition in the model.

Partitioned models can be defined with values explicitly named in the Value Domain field of a partitioning attribute, or that field can be left blank. If you specify explicit values for any model partitioning attribute in the Value Domain field, this causes problems for some Decision Center reports. For more information, see Section 8.1, "Multiple Rows Appearing for Some Reports Based on Partitioned Model Data."

The workaround for this issue is *not* to fill in the Value Domain field for any partitioning attribute.

# 8 Oracle RTD Decision Center Issues

This section provides release notes for Oracle RTD Decision Center. It contains the following topic:

- Section 8.1, "Multiple Rows Appearing for Some Reports Based on Partitioned Model Data"

## 8.1 Multiple Rows Appearing for Some Reports Based on Partitioned Model Data

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

When an Oracle RTD model is defined with partitions, then at run-time, when each new partitioning attribute value is presented to the learning process, Oracle RTD automatically creates a new partition in the model.

When any partitioning attribute in a model has one or more explicitly defined values in its Value Domain in the Inline Service, this causes the following problems in Decision Center reports that can filter on the partitioning attributes of that model:

- The partitioning attribute does not appear among the filtering parameters in the report header, so users cannot select a particular attribute value to filter on

- Multiple rows appear for other input attributes, instead of one per attribute

> **Note:** This problem does not arise if values are *not* explicitly defined for a partitioning attribute in the Inline Service model. In this case, Decision Center reports display the partitioning attribute field in the report header, enabling users to select a value from the field's drop-down list.

The image that follows shows an example of this issue, as applied to the Inline Service **CrossSellwithpartition**, and to a subsequent Decision Center report. In the Inline Service, the partitioning attribute Session/Channel for the Offer Acceptance model has two explicitly defined attribute values, "Call" and "Web", in its Value Domain.

Assuming both of the attribute values, "Call" and "Web", have been passed to the model learning process, the effects of the explicit attribute definition are shown on an Analysis Drivers report in the Decision Center, as follows:

- The filter parameter fields in the report header should include the field "Channel", but that field is not present in the report header

- In the report body, two rows appear for each of the customer attributes Age, Amount of Pending Transactions, AvailableCreditAsPercentOfCreditLine, and CallReason
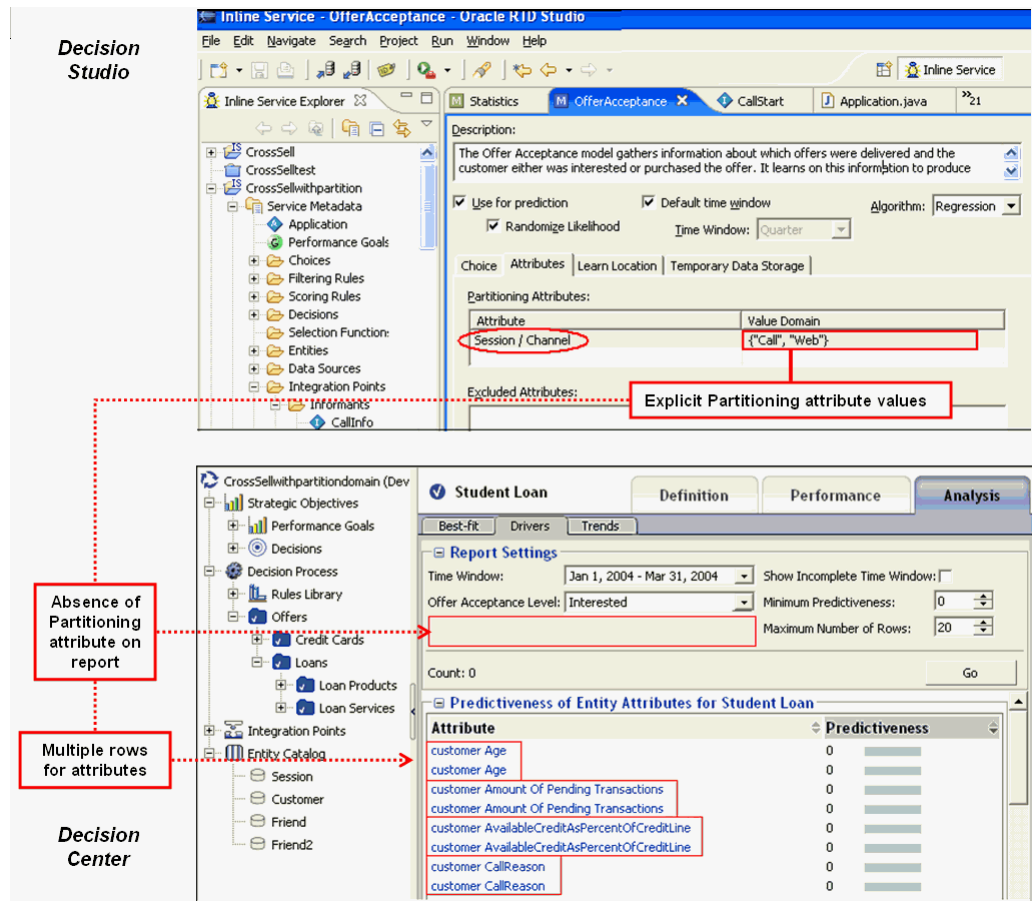
The workaround for this issue is *not* to fill in the Value Domain field for any partitioning attribute in the Inline Service.

# 9 Load Generator Issues

This section provides release notes for Oracle RTD Load Generator. It contains the following topics:

- Section 9.1, "Load Generator Client Errors Are Logged to server.log Instead of client.log"

- Section 9.2, "Load Generator Help Search Button Does Not Work"

- Section 9.3, "Load Generator Help Home Button Does Not Work"

## 9.1 Load Generator Client Errors Are Logged to server.log Instead of client.log

This issue applies to Version 2.2 only.

By default, Load Generator logs client-side errors to the `server.log` file. These errors should instead be logged to the `client.log` file.

To work around this problem, update the file *RTD_HOME*`\etc\sdconfig-machine.xml` on the computer where Load Generator is running, so that `SDLoggingFileName` is set to `client.log`, as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<properties:SDConfigType xmlns:properties="http://www.sigmadynamics.com/
schema/properties">
  <properties:category name="Logging">
    <schema:property internalName="SDLoggingFileName"
default="RTD_HOME/log/client.log" visibility="false" modifiable="true"
type="String" clusterWide="false" />
  /properties:category>
</properties:SDConfigType>
```

Make sure to replace *RTD_HOME* with the actual Oracle RTD installation path.

> **Note:** *RTD_HOME* refers to the Oracle RTD installation directory, not
> the *RTD_RUNTIME_HOME* directory that is located with your
> application server installation files. To ensure you are updating the
> correct file in the correct directory, follow these steps:
>
> 1. Locate the file *RTD_HOME*\scripts\loadgen.cmd.
> 2. From the directory where loadgen.cmd resides, go to
>    ..\etc\sdconfig-machine.xml.
> 3. Update the file as described in the previous instructions.

## 9.2 Load Generator Help Search Button Does Not Work

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

For Oracle RTD Version 2.2, the Load Generator Help Search button does not work.
This issue has no workaround.

## 9.3 Load Generator Help Home Button Does Not Work

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

For Oracle RTD Version 2.2, the Load Generator Help Home button does not work.
This issue has no workaround.

# 10 Globalization Issues

This section provides release notes related to Oracle RTD globalization. It contains the
following topic:

- Section 10.1, "Oracle RTD User Interfaces Only Available in English"

## 10.1 Oracle RTD User Interfaces Only Available in English

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

Although Oracle RTD is a localized product, its user interfaces are currently only
available in English. In addition, there are several known issues with Decision Center
reports relating to locale-sensitive data, including incorrect display format for date,
time, and currency. These issues will be addressed in future releases.

# 11 Documentation Errata

This section provides information about errata in Oracle RTD documentation, in particular, *Oracle Real-Time Decisions Installation and Administration of Oracle RTD*, which applies to Versions 2.2, 2.2.1, and 2.2.1.1.

This section contains the following topics:

## 11.1 Oracle RTD Documentation Does Not Provide Information About Setting Up JConsole for SSL

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

*Oracle Real-Time Decisions Installation and Administration of Oracle RTD* does not provide information about how to set up JMX MBean access in a secure fashion. Refer to the documentation for your J2EE container for information about how to set up JConsole for SSL.

## 11.2 Additional Information Available on Setting Up SSL-Only Configuration for Oracle RTD on Oracle Application Server

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

For greater security, when setting up SSL, you should disable the regular HTTP port to ensure that all client connections are routed through the SSL port.

When Oracle RTD is deployed on Oracle Application Server, you may need to perform the following additional steps when setting up an SSL-only configuration:

1. Run the following keytool command to import the Oracle RTD certificate into the Oracle Application Server cacerts:

   ```
   keytool -import -file RTD_HOME/etc/ssl/sdserver.cer -keystore cacerts
   ```

   By default, `keytool` is located in `ORACLE_AS_HOME/jdk/bin/keytool.exe`.

2. When prompted, enter the keystore password for the application server's cacerts file.

The default location of cacerts is *ORACLE_AS_HOME*/jdk/jre/lib/security/cacerts.

This procedure is useful for test environments. For production systems, you should use your own certificate rather than the test certificate that ships with Oracle RTD.

## 11.3  Updated startJConsole.bat Example Available

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

In *Oracle Real-Time Decisions Installation and Administration of Oracle RTD*, Section 4.7.2, Setting up a batch file for JConsole, the existing information does not allow for cases where the *WEBSPHERE_HOME* path contains spaces.

To ensure that the script executes correctly when *WEBSPHERE_HOME* contains spaces, use the following updated lines to create the startJConsole.bat file:

```
set WAS_HOME=WEBSPHERE_HOME\AppServer
set USER_HOME=WEBSPHERE_HOME\AppServer\profiles\profile_name
set WAS_HOST=localhost
set WAS_BOOTSTRAP_PORT=jmx_remote_port

"%WAS_HOME%\java\bin\jconsole" -J-Djava.class.path="%WAS_
HOME%\runtimes\com.ibm.ws.admin.client_6.1.0.jar;%WAS_
HOME%\java\lib\tools.jar" -J-Dcom.ibm.CORBA.ConfigURL="file:%USER_
HOME%\properties\sas.client.props" -J-Dcom.ibm.SSL.ConfigURL="file:%USER_
HOME%\properties\ssl.client.props" service:jmx:iiop://%WAS_HOST%:%WAS_
BOOTSTRAP_PORT%/jndi/JMXConnector
```

For *jmx_remote_port*, enter the JConsole port number. For WebSphere, the JMX remote port is always the same as the WebSphere bootstrap port. Make sure to replace *WEBSPHERE_HOME* with the actual WebSphere installation path, and replace *profile_name* with the name of your WebSphere profile. For example:

```
set WAS_HOME=C:\Program Files\IBM\WebSphere\AppServer
set USER_HOME=C:\Program Files\IBM\WebSphere\AppServer\profiles\AppSrv01
set WAS_HOST=localhost
set WAS_BOOTSTRAP_PORT=12345

"%WAS_HOME%\java\bin\jconsole" -J-Djava.class.path="%WAS_
HOME%\runtimes\com.ibm.ws.admin.client_6.1.0.jar;%WAS_
HOME%\java\lib\tools.jar" -J-Dcom.ibm.CORBA.ConfigURL="file:%USER_
HOME%\properties\sas.client.props" -J-Dcom.ibm.SSL.ConfigURL="file:%USER_
HOME%\properties\ssl.client.props" service:jmx:iiop://%WAS_HOST%:%WAS_
BOOTSTRAP_PORT%/jndi/JMXConnector
```

> **Tip:** Ensure that the JConsole command at the end of this file
> ("%WAS_HOME%\java\bin\jconsole...") is all on one line.

## 11.4  Increasing JVM Memory for Oracle RTD on OC4J

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

*Oracle Real-Time Decisions Installation and Administration of Oracle RTD* provides incorrect information about how to increase the JVM memory when Oracle RTD is deployed on OC4J. The correct instructions are as follows:

To increase the JVM memory when Oracle RTD is running on Windows, add the following line to the oc4j.cmd file:

```
set JVMARGS=%JVMARGS% -Xms512M -Xmx1024M
```

To increase the JVM memory when Oracle RTD is running on UNIX, add the following
line to the `oc4j` file:

```
JVMARGS=$JVMARGS -Xms512M -Xmx1024M
```

## 11.5  Setting PermGen Size for Oracle RTD on OC4J

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

During OC4J setup, explicitly specify the PermGen size in the JVM arguments, in
order to decrease out-of-memory issues when deploying an Inline Service.

The section Configuring Server Properties for Standalone OC4J in *Oracle Real-Time
Decisions Installation and Administration of Oracle RTD* describes how to configure the
JVM arguments.

To specify the PermGen size, add the PermSize parameter to one of the `set JVMARGS`
lines in the `oc4j.cmd` file, as in the following example:

```
set JVMARGS=%JVMARGS% -Xms512M -Xmx1024M -XX:PermSize=256m
```

## 11.6  Setting the Protocol Attribute for rtd-web-site.xml on OC4J

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

*Oracle Real-Time Decisions Installation and Administration of Oracle RTD* provides
information about setting up `rtd-web-site.xml` when Oracle RTD is running on
OC4J. However, these instructions do not address how to set the `protocol` attribute.
If the `protocol` attribute is set to `ajp13`, and another Web site exists for that OC4J
instance that is also using the `ajp13` protocol, you may encounter problems.

To avoid this issue, make sure to set the `protocol` attribute in `rtd-web-site.xml`
to `http`, as follows:

```
<?xml version="1.0"?>
<web-site xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://xmlns.oracle.com/oracleas/schema/
web-site-10_0.xsd" port="8080" protocol="http" display-name="Oracle RTD Web Site"
schema-major-version="10" schema-minor-version="0" >
<default-web-app application="default" name="defaultWebApp" root="/j2ee" />
<access-log path="../log/rtd-web-access.log" split="day" />
</web-site>
```

## 11.7  Adding Additional JDBC Data Sources on OC4J

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

*Oracle Real-Time Decisions Installation and Administration of Oracle RTD*, Section 7.1.3,
Step 2 provides incorrect information about editing the `orion-web.xml` files to add
an additional JDBC data source on OC4J.

The incorrect instructions state that you should edit the following files:

```
./rtis/WEB-INF/orion-web.xml
./soap/WEB-INF/orion-web.xml
```

Instead, you should edit these files:

```
./rtis/orion-web.xml
./soap/orion-web.xml
```

## 11.8  Private Key Alias is Incorrect

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

*Oracle Real-Time Decisions Installation and Administration of Oracle RTD*, Section 5.6, Step 6b provides incorrect information about the Private Key Alias.

The incorrect instructions are as follows:

- For **Private Key Alias**, **Private Key Passphrase**, and **Confirm Private Key Passphrase**, enter the password for your keystore. If you are using the default Oracle Real-Time Decisions keystore, enter the password you created in Section 2.4.

The correct instructions are as follows:

- For **Private Key Alias**, enter `tc-ssl`.

  For **Private Key Passphrase** and **Confirm Private Key Passphrase**, enter the password for your keystore. If you are using the default Oracle Real-Time Decisions keystore, enter the password you created in Section 2.4.

## 11.9  SSL Website Incorrect

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

*Oracle Real-Time Decisions Installation and Administration of Oracle RTD*, Section 5.6.1, provides incorrect information about the exact URL to call to verify that the SSL port is functioning properly.

The incorrect URL is:

```
http://server_name:ssl_port/ui
```

The correct URL is:

```
https://server_name:ssl_port/ui
```

## 11.10  Specifying IPV4 Preference in JVM Startup for All Unix Systems

This issue applies to Version 2.2, Version 2.2.1, and Version 2.2.1.1.

In the Oracle Real-Time Decisions Installation and Administration of Oracle RTD, for each of the application servers, OC4J, WebSphere, and WebLogic, there is an incorrect qualification for one of the configuration setup steps.

In each case, the particular configuration step indicates that you must specify version 4 of the internet protocol (IP) for the JVM startup, rather than any other version, using the following parameter:

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

The incorrect qualification for each of these steps is:

```
If you installed application_server on an AIX-based system
```

The correct qualification for each of these steps is:

```
If you installed application_server on any Unix-based system
```

## 11.11  Configuration Properties WebServerPort and HTTPSEnabled Not Required in Version 2.2.1 Configuration Procedures

This issue applies to Versions 2.2.1 and 2.2.1.1 only.

In *Oracle Real-Time Decisions Installation and Administration of Oracle RTD*, for each of the application servers, OC4J, WebSphere, and WebLogic, the setup steps referring to the configuration properties WebServerPort and HTTPSEnabled are superfluous in Versions 2.2.1 and 2.2.1.1.

This affects the description of the following two topics, which appear separately for each application server:

- Configuring SSL for Real-Time Decision Server
- Changing the Oracle Real-Time Decisions Port Number

### Configuring SSL for Real-Time Decision Server

For each of the application servers, OC4J, WebSphere, and WebLogic, the penultimate step in this topic includes a Note ("For a truly secure environment..."). This Note includes two steps for you to execute. In configuring SSL for Oracle RTD Version 2.2.1 or Oracle RTD Version 2.2.1.1, you should *not* execute Step 1 ("Use JConsole to change...").

### Changing the Oracle Real-Time Decisions Port Number

For each of the application servers, OC4J, WebSphere, and WebLogic, you can change the Oracle RTD application port number. Prior to Version 2.2.1, this was a two-stage process, consisting of changing the port number in both the application server *and* in Oracle RTD. With Versions 2.2.1 and 2.2.1.1, the second stage (changing the port number in Oracle RTD) is not required, and should not be executed.

# 12  Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. 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.

**Deaf/Hard of Hearing Access to Oracle Support Services**

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at http://www.fcc.gov/cgb/consumerfacts/trs.html, and a list of phone numbers is available at http://www.fcc.gov/cgb/dro/trsphonebk.html.