

Oracle® Real-Time Decisions Base Application
Decision Management Installation and Configuration Guide
Release 3.1
E19023-01

March 2011

Oracle Real-Time Decisions Base Application Decision Management Installation and Configuration Guide,
Release 3.1

E19023-01

Copyright © 2011, Oracle and/or its affiliates. All rights reserved.

Primary Author: Oracle Corporation

Contributors: Oracle Real-Time Decisions development, product management, and quality assurance teams.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	v
----------------------	---

1 Installing Oracle RTD Decision Management

1.1	Installation Overview	1-1
1.2	Installing Oracle RTD Decision Management for Development	1-1
1.2.1	Preparing for Installation.....	1-2
1.2.1.1	Installing Oracle JDeveloper	1-2
1.2.1.2	Setting Up Oracle RTD and the Oracle RTD Database	1-2
1.2.1.3	Creating the WebLogic Domain	1-3
1.2.1.4	Configuring Oracle RTD	1-4
1.2.2	Installing Oracle RTD Decision Management	1-5
1.2.2.1	Unzipping the Oracle RTD Decision Management Product.....	1-5
1.2.2.2	Oracle RTD Decision Management Database Creation	1-5
1.2.2.3	Oracle RTD Decision Management Data Source Setup	1-6
1.2.2.4	Oracle RTD Decision Management Data Source Access by Oracle RTD Setup..	1-8
1.2.2.5	Inline Service Deployment	1-8
1.2.2.6	Storing Credentials to Make Oracle RTD Web Service Calls from Oracle RTD Decision Management	1-8
1.2.2.7	Oracle RTD Decision Management Application Deployment.....	1-9
1.3	Preparing Your Development Application for Production	1-10
1.4	Installing Oracle RTD Decision Management for Production	1-11
1.4.1	Preparing for Installation.....	1-11
1.4.2	Production WebLogic Domain Creation.....	1-11
1.4.3	Installing Oracle RTD Decision Management	1-14
1.4.3.1	Files Required from Development Environment.....	1-15
1.4.3.2	Oracle RTD Decision Management Database Creation	1-15
1.4.3.3	Oracle RTD Decision Management Data Source Setup	1-16
1.4.3.4	Oracle RTD Decision Management Data Source Access by Oracle RTD Setup	1-17
1.4.3.5	Storing Credentials to Make Oracle RTD Web Service Calls from Oracle RTD Decision Management	1-18
1.4.3.6	Oracle RTD Decision Management Application Deployment.....	1-19
1.4.3.7	Inline Service Deployment	1-20
1.5	Security Configuration	1-20
1.5.1	Managing Users Within JDeveloper in a Development Environment	1-20
1.5.1.1	Retrieving the Encrypted Password for Your User	1-21

1.5.1.2	Adding Users in Oracle RTD Decision Management Metadata and Mapping Them to Decision Management Application Roles	1-21
1.5.1.3	Running Application Generation	1-22
1.5.1.4	Deploying the Oracle RTD Decision Management Application	1-22
1.5.1.5	Adding Users to Oracle RTD Groups	1-22
1.5.2	Mapping Enterprise Users and Groups to Decision Management Application Roles in a Development Environment	1-23
1.5.2.1	Adding the Enterprise Role in Oracle RTD Decision Management Metadata and Mapping it to Decision Management Application Roles	1-23
1.5.2.2	Running Decision Management Application Generation	1-23
1.5.2.3	Deploying the Decision Management Application	1-24
1.5.2.4	Specify the WebLogic Group as a Member of Oracle RTD Groups	1-24
1.5.2.5	Mapping Your Enterprise Users or Groups to the Oracle RTD Decision Management Group	1-24
1.5.3	Mapping Enterprise Users and Groups to Decision Management Application Roles in a Production Environment	1-24
1.5.3.1	Mapping Users or Groups to Decision Management Application Roles	1-25
1.5.3.2	Adding the User or Group to Oracle RTD Groups	1-25
1.5.4	Using SSL with Oracle RTD Decision Management Applications	1-25
1.5.4.1	SSL for Development Environment	1-26
1.5.4.2	SSL for Production Environment	1-27

2 Configuring Oracle RTD Decision Management

2.1	Oracle RTD Decision Management Architecture Overview	2-1
2.2	Configuring Oracle RTD Decision Management	2-3
2.2.1	Overview	2-3
2.2.2	Oracle RTD Decision Management Metadata Configuration Files	2-4
2.2.2.1	Config XML File	2-6
2.2.2.2	Perspectives XML File	2-6
2.2.2.3	Choice Group and Relationship-Types XML Files	2-12
2.2.2.4	Security XML Files	2-16
2.2.2.5	Folder for Choice Group Images	2-17
2.2.2.6	Inline Service Folder	2-17
2.2.3	Ant Tasks	2-18
2.2.4	Java API	2-19
2.2.5	Application Extensions	2-19
2.2.5.1	Adding an Attribute to a Choice Group	2-19
2.2.5.2	Adding a Choice Group	2-19
2.2.5.3	Creating a Relationship Type Between Two Choice Groups	2-20
2.2.5.4	Modifying Perspectives	2-20
2.3	Miscellaneous	2-20
2.3.1	Configuring Logs	2-20

Preface

This document describes the installation and configuration of Oracle Real-Time Decisions (Oracle RTD) Decision Management applications.

Audience

This document is intended for the following Oracle RTD users:

- Technical users configuring Oracle RTD Decision Management applications using Decision Designer
- Administrators

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.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/support/contact.html> or visit <http://www.oracle.com/accessibility/support.html> if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Real-Time Decisions platform version 3.0 documentation set and the Oracle Real-Time Decisions Base Application Release 3.1 documentation set:

- *Oracle Real-Time Decisions Installation and Administration Guide*
- *Oracle Real-Time Decisions Platform Developer's Guide*
- *Oracle Real-Time Decisions Decision Center User's Guide*
- *Oracle Real-Time Decisions Release Notes*
- *Oracle Real-Time Decisions Base Application Installation and Reference Guide*
- *Oracle Real-Time Decisions Base Application Decision Management Applications User's Guide*

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.
<i>italic</i>	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.

Installing Oracle RTD Decision Management

Terminology: The term "reference implementation" is used in this chapter to refer to the specific Oracle RTD Decision Management application Oracle RTD for Marketing Optimization (also referred to as the RTD for Marketing Optimization application) released with Oracle RTD Base Application.

This chapter contains the following topics:

- [Section 1.1, "Installation Overview"](#)
- [Section 1.2, "Installing Oracle RTD Decision Management for Development"](#)
- [Section 1.3, "Preparing Your Development Application for Production"](#)
- [Section 1.4, "Installing Oracle RTD Decision Management for Production"](#)
- [Section 1.5, "Security Configuration"](#)

1.1 Installation Overview

The steps that lead to the deployment of a Oracle RTD Decision Management application are as follows:

- Using Oracle JDeveloper, a developer configures the Oracle RTD Decision Management application and deploys it to the Weblogic Server that comes with JDeveloper.
- After the application has been properly configured, the developer creates an Oracle RTD Decision Management application EAR file.
- The domain administrator deploys the EAR file to a remote Oracle WebLogic Server using Enterprise Manager to be used in the production (or test) environment.

Summarizing, there are two environments:

- Development environment - the environment for the configuration of Oracle RTD Decision Management
- Production environment - the environment for the use of the configured Oracle RTD Decision Management application either for testing or production

1.2 Installing Oracle RTD Decision Management for Development

This section contains the following topics:

- [Section 1.2.1, "Preparing for Installation"](#)
- [Section 1.2.2, "Installing Oracle RTD Decision Management"](#)

1.2.1 Preparing for Installation

The outline steps to perform in preparing to install Oracle RTD Decision Management are as follows:

1. Install Oracle Database.
2. Install Oracle JDeveloper 11g Rel 1.
3. Install Oracle RTD for WebLogic, then create and initialize the Oracle RTD database.
4. Create a WebLogic domain with an Administration Server only.
5. Configure Oracle RTD to run on the Administration Server.

The rest of this section contains the following topics:

- [Section 1.2.1.1, "Installing Oracle JDeveloper"](#)
- [Section 1.2.1.2, "Setting Up Oracle RTD and the Oracle RTD Database"](#)
- [Section 1.2.1.3, "Creating the WebLogic Domain"](#)
- [Section 1.2.1.4, "Configuring Oracle RTD"](#)

1.2.1.1 Installing Oracle JDeveloper

To install Oracle JDeveloper 11g Rel 1, perform the following steps:

1. In JDeveloper:
 - Select Tools, then Preferences.
 - Select the Environment tab and set the Encoding to UTF-8.
 - Click OK.
2. Create the following environment variables:
 - `ANT_HOME=<Oracle Middleware Install dir>\jdeveloper\ant`
 - `JAVA_HOME=<Oracle Middleware Install dir>\jdk160_21`Add `%ANT_HOME%\bin` and `%JAVA_HOME%\bin` to your PATH.
3. Create a file called `clm-build.properties` in your home directory (on Windows, this folder can be reached using the variable `%HOMEPATH%`). This file should contain:
 - `jdeveloper.home=<Oracle Middleware Install dir>`For example, `C:/Oracle/Middleware`.

1.2.1.2 Setting Up Oracle RTD and the Oracle RTD Database

To set up Oracle RTD and the Oracle RTD database, install Oracle RTD for WebLogic, then create and initialize the Oracle RTD database, as described in Sections 2.1 and 2.2 of *Oracle Real-Time Decisions Installation and Administration Guide*.

Note: In addition to the instructions in the specified sections, you must make the following extra provision for the Oracle RTD database SDDB:

- Before you can initialize the database either by using SDDBTool or by running the database initialization command line scripts, you must have the system privilege CREATE VIEW.
-
-

1.2.1.3 Creating the WebLogic Domain

The WebLogic domain to create for the development environment will have the following properties:

- The domain will be configured to support Oracle JRF
- The domain will have only one Administration Server

To create the development WebLogic domain, perform the following steps:

1. Start the Configuration Wizard.

On Windows, this is available via Start Menu: Programs -> Oracle Fusion Middleware <11g_version> -> WebLogic Server 11gR1 -> Tools -> Configuration Wizard

2. In the Welcome window, select **Create a new WebLogic domain**, then click Next.
3. In the Select Domain Source window:
 - Select **Generate a domain configured automatically to support the following products**.
 - Select the **Oracle JRF - 11.1.1.0 [oracle_common]** option.
 - Click Next.
4. In the Specify Domain Name and Location, specify a name and location for your domain, then click Next.

Note: You can specify any name for the domain. The generic name name <RTDCLM_Dev_Domain> will be used to refer to the development environment domain name in all the setup steps of this and other sections.

5. In the Configure Administrator User Name and Password window, specify a user name and password for the administrator account, then click Next.
6. In the Configure Server Start Mode and JDK window, keep the default values in the **WebLogic Domain Startup Mode** and **Available JDKs** areas, then click Next.
7. In the Select Optional Configuration window, for the development environment select only **Administration Server**, then click Next.
8. In the Configure the Administration Server window:
 - Keep the default values for Name and Listen address.
 - For Listen port, either keep the default value of 7001 or enter the port number that you require for the Administration Server.
 - Click Next.

9. Review the domain configuration details in the Configuration Summary window, then click Create.

1.2.1.4 Configuring Oracle RTD

This section describes the configuring of Oracle RTD to run on the Administration Server.

-
- Notes:**
1. The rationale for running Oracle RTD on the Administration Server in the development environment is to use less memory. However this should never be used in a production environment.
 2. If you want, you can configure Oracle RTD to run on a Managed Server in your development environment. Note that *Oracle Real-Time Decisions Installation and Administration Guide* does not describe how to configure Oracle RTD to run on an Administration Server.
-

With reference to the instructions in chapter 5 of the *Oracle Real-Time Decisions Installation and Administration Guide*, perform the following steps:

1. If you are running Oracle RTD in a Managed Server, perform the instructions in Section 5.4.

If you are running Oracle RTD in the Administration Server instead of a Managed Server, replace Section 5.4 with these instructions:

For **Windows**: In the bin subfolder of your domain, edit `startWebLogic.cmd` and add the following entries at the top of the file:

```
set JAVA_OPTIONS=%JAVA_OPTIONS%
-Dorg.eclipse.emf.ecore.EPackage.Registry.INSTANCE=com.sigmadynamics.emf.util.S
DEMFRegistry
set JAVA_OPTIONS=%JAVA_OPTIONS%
-Djavax.xml.parsers.SAXParserFactory=com.sun.org.apache.xerces.internal.jaxp.SA
XParserFactoryImpl
set JAVA_OPTIONS=%JAVA_OPTIONS% -Dcom.sun.management.jmxremote=true
set JAVA_OPTIONS=%JAVA_OPTIONS% -Dcom.sun.management.jmxremote.port=12346
set JAVA_OPTIONS=%JAVA_OPTIONS%
-Dcom.sun.management.jmxremote.authenticate=false
set JAVA_OPTIONS=%JAVA_OPTIONS% -Dcom.sun.management.jmxremote.ssl=false
set JAVA_OPTIONS=%JAVA_OPTIONS% -Dweblogic.wsee.skip.async.response=true
```

For **Unix**: In the bin subfolder of your domain, edit `startWebLogic.sh` and add at the top of the file:

```
JAVA_OPTIONS="{JAVA_OPTIONS}"
-Dorg.eclipse.emf.ecore.EPackage.Registry.INSTANCE=com.sigmadynamics.emf.util.S
DEMFRegistry"
JAVA_OPTIONS="{JAVA_OPTIONS}"
-Djavax.xml.parsers.SAXParserFactory=com.sun.org.apache.xerces.internal.jaxp.SA
XParserFactoryImpl"
JAVA_OPTIONS="{JAVA_OPTIONS}" -Dcom.sun.management.jmxremote=true"
JAVA_OPTIONS="{JAVA_OPTIONS}" -Dcom.sun.management.jmxremote.port=12346"
JAVA_OPTIONS="{JAVA_OPTIONS}"
-Dcom.sun.management.jmxremote.authenticate=false"
JAVA_OPTIONS="{JAVA_OPTIONS}" -Dcom.sun.management.jmxremote.ssl=false"
JAVA_OPTIONS="{JAVA_OPTIONS}" -Dweblogic.wsee.skip.async.response=true"
JAVA_OPTIONS="{JAVA_OPTIONS}" -Djava.net.preferIPv4Stack=true"
```

2. Do *not* perform the steps in Section 5.5.

3. If you are running Oracle RTD in a Managed Server, perform the instructions as they appear from Section 5.6 to the end of Chapter 5.

If you are running Oracle RTD in the Administration Server instead of a Managed Server, perform the operations from Sections 5.6 to the end of Chapter 5 on the Administration Server instead of the Managed Server.

1.2.2 Installing Oracle RTD Decision Management

After you have created the WebLogic domain and configured Oracle RTD to run on the Administration Server, you are ready to install Oracle RTD Decision Management.

Note: This section describes how to install Oracle RTD Decision Management and the specific RTD for Marketing Optimization application, as released with Oracle RTD Base Application. For setting up other Oracle RTD Decision Management applications, see [Section 2.2, "Configuring Oracle RTD Decision Management."](#)

This section contains the following topics:

- [Section 1.2.2.1, "Unzipping the Oracle RTD Decision Management Product"](#)
- [Section 1.2.2.2, "Oracle RTD Decision Management Database Creation"](#)
- [Section 1.2.2.3, "Oracle RTD Decision Management Data Source Setup"](#)
- [Section 1.2.2.4, "Oracle RTD Decision Management Data Source Access by Oracle RTD Setup"](#)
- [Section 1.2.2.5, "Inline Service Deployment"](#)
- [Section 1.2.2.6, "Storing Credentials to Make Oracle RTD Web Service Calls from Oracle RTD Decision Management"](#)
- [Section 1.2.2.7, "Oracle RTD Decision Management Application Deployment"](#)

1.2.2.1 Unzipping the Oracle RTD Decision Management Product

Unzip `RTD_Designer.zip` from the Oracle RTD Base Application distribution into the directory of your choice.

Note: Some zip extraction utilities impose a limit on the combined length of the target directory path and the full directory+file name in the zip file. Oracle recommends that select your target directory with this in mind when you unzip `RTD_Designer.zip`.

1.2.2.2 Oracle RTD Decision Management Database Creation

Note: For running the SQL commands and SQL scripts specified in this section, use SQL*Plus or an equivalent tool that can run on your database, such as the Database Navigator of Oracle JDeveloper.

To create an Oracle RTD Decision Management database, perform the following steps:

1. Create a database user for the Oracle RTD Decision Management schema.

```
CREATE USER <username> IDENTIFIED BY <password> DEFAULT
TABLESPACE users TEMPORARY TABLESPACE temp QUOTA UNLIMITED ON
users;
```

```
GRANT CREATE VIEW, CONNECT, RESOURCE TO <username>;
```

Note: If you are going to use non 7-bit ASCII characters in choice group id or choice attribute ids, you must set the environment variable NLS_LANG to .UTF8 before running the sql scripts in the steps that follow.

2. Create the schema, by performing the following steps:

a. *Preliminary step if you want to use Oracle JDeveloper:*

Open Oracle JDeveloper, open Database Navigator, then connect as the user that you just created.

b. Go to the subfolder **clm\Database\sql** of where you unzipped RTD_Designer.zip, then run the file `load_core.sql`.

Note: The file `load_core.sql` contains generic commands, required for all Oracle RTD Decision Management applications. The commands in `load_ils.sql`, as released, are for the RTD for Marketing Optimization application.

c. Go to the subfolder **clm\Database\sql\ils** of where you unzipped RTD_Designer.zip, then run the file `load_ils.sql`.

Make sure you have committed your changes.

Note: The following step is specific to the installation of the application RTD for Marketing Optimization, as released with Oracle RTD Base Application. The step seeds the database with some sample data.

3. Load the sample seed data into the Oracle RTD Decision Management database for the reference implementation application, by performing the following steps:

- Run `clm\Build\metadata\ref\sql\insert_ils_data.sql` first to create some channels, placements, slots and slot types.
- Run `clm\Build\metadata\ref\sql\insert_marketing_data.sql` to create some campaigns, offers and creatives.

You may notice the `drop_core.sql` and `drop_ils.sql` files. These can be used to drop the schema. Call `drop_ils.sql` first, then `drop_core.sql`.

1.2.2.3 Oracle RTD Decision Management Data Source Setup

To set up an Oracle RTD Decision Management data source, perform the following steps:

1. Start the Administration Server.

On Windows, this is available via the Start Menu: Programs -> Oracle Fusion Middleware <11g_version> -> User Projects -> <RTDCLM_Dev_Domain> -> Start Admin Server for Weblogic Server Domain

2. Open the Admin Server Console.

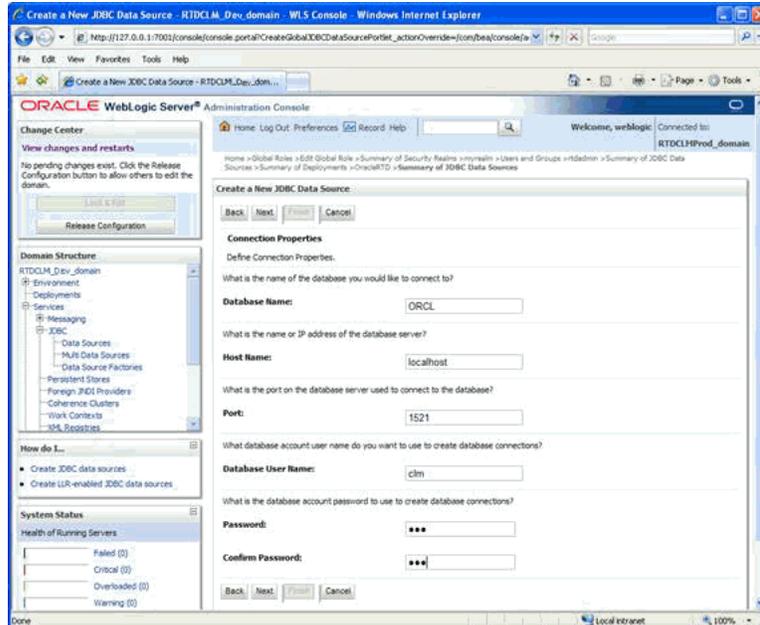
On Windows, this is available via the Start Menu: Programs -> Oracle Fusion Middleware <11g_version> -> User Projects -> <RTDCLM_Dev_Domain> -> Admin Server Console.

3. Log in with the administrator username and password, which was specified during domain creation (see [Section 1.2.1.3, "Creating the WebLogic Domain."](#)).
4. Navigate the path Services -> Data Sources -> New -> Generic Data Source, then enter:

- Name: **CLM_DS**
- JNDI Name: **CLMDS**
- Database Type: **Oracle**

Click Next.

5. For the Database Driver, select **Oracle's Driver (Thin) for Instance connections; Versions: 9.0.1 and later**, then click Next.
6. In the Transaction Options window, deselect **Supports Global Transactions**, then click Next.
7. For the Connection Properties:
 - Specify the database settings that match your configuration, for example:



Ensure that the Database User name and Password match the values that you set up in step 1 of [Section 1.4.3.2, "Oracle RTD Decision Management Database Creation."](#)

- Click Next.
8. Leave all the settings already filled, except enter CHOICE for the Test Table Name, and click Test Configuration. Then click Next.

9. Select AdminServer as a target, then click Finish.

1.2.2.4 Oracle RTD Decision Management Data Source Access by Oracle RTD Setup

Section 8.3.3 of *Oracle Real-Time Decisions Installation and Administration Guide* describes the general steps to add a data source to Oracle RTD. Follow the steps in that section, and, specifically for the Oracle RTD Decision Management data source, add the following in `web.xml` for `rtis.war`:

```
<resource-ref id="CLM_RTIS">
  <res-ref-name>CLMDS</res-ref-name>
  <res-type>javax.sql.DataSource</res-type>
  <res-auth>Container</res-auth>
  <res-sharing-scope>Unshareable</res-sharing-scope>
</resource-ref>
```

and add the following in `web.xml` for `soap.war`:

```
<resource-ref id="CLM_Axis">
  <res-ref-name>CLMDS</res-ref-name>
  <res-type>javax.sql.DataSource</res-type>
  <res-auth>Container</res-auth>
  <res-sharing-scope>Unshareable</res-sharing-scope>
</resource-ref>
```

1.2.2.5 Inline Service Deployment

To deploy the Oracle RTD Decision Management application, you must have a specific Inline Service running in the Oracle RTD instance on the same server as Oracle RTD Decision Management. You need to deploy that Inline Service to the Oracle RTD server using Decision Studio.

By convention, the Inline Service is located in a sub-folder called `service` of the application module folder. For the reference implementation, that folder is `clm\Build\metadata\ref\service`.

1.2.2.6 Storing Credentials to Make Oracle RTD Web Service Calls from Oracle RTD Decision Management

Oracle RTD Decision Management makes web service calls from the Decision Management server side to Oracle RTD. Credentials for these calls are stored in the WebLogic credential store. In summary form, the credential setup process is as follows:

- First, you create a user that will be used to authenticate the web service calls
- Then, you store the credentials of this user in the WebLogic credential store, so that the Decision Management application can retrieve them

To create the user, perform the following steps:

1. Open the Admin Server Console.

On Windows, this is available via the Start Menu: Programs -> Oracle Fusion Middleware <11g_version> -> User Projects -> <RTDCLM_Dev_Domain> -> Admin Server Console.
2. Log in with the administrator username and password.
3. Select Security Realms and select the security realm you are using (myrealm by default).

4. Select the Users and Groups tab, and click New to create a new user.
5. Specify a username and password (generic names `<clmclient_user>` and `<clmclient_password>`).
6. Click OK.
7. Select the user that you just created and select the Groups tab.
8. Select the RTDDCUserGroup group, shuttle it to the right.
9. Click Save:

To store the username and password in the WebLogic credential store, perform the following steps:

1. In a command window, go to the directory `<middleware_home>\oracle_common\common\bin\`, and run `wlst.cmd` on Windows or `wlst.sh` on Unix.

Note: `<middleware_home>\wlserver_10.3\common\bin` includes similarly named command files - do *not* execute those.

2. Using the values appropriate to your environment (specified during domain creation), type the following command:

```
connect(' <admin_user>', ' <admin_password>', ' <server_name>: <port>')
```

3. Using the same username and password that you created previously in this section, enter:

```
updateCred(map="oracle.rtd.clm", key="clm.client",
user=" <clmclient_user>", password=" <clmclient_password>")
```

4. Verify that this worked, by entering:

```
listCred(map="oracle.rtd.clm", key="clm.client")
```

5. Complete the process by entering:

```
exit()
```

1.2.2.7 Oracle RTD Decision Management Application Deployment

Deploy the Oracle RTD Decision Management application in JDeveloper, by performing the following steps:

1. Open JDeveloper from Start Menu, Programs -> Oracle Fusion Middleware `<11g_version>` -> JDeveloper Studio `<11g_version>`.

If prompted, select Default Role.

2. Click Open Application... and select `clm.jws` in the `clm` subfolder of where you unzipped `RTD_Designer.zip`.
3. Select Application -> Deploy -> `clm_application1...` and, in the Deployment Action window, select **Deploy to Application Server**.
4. At top right of the Select Server window, click the green plus button.

The Create Application Server Connection process that opens up has five steps:

- Name and Type: Select AdminServer and WebLogic 10.3.
- Authentication: Specify the administrator username and password.

- Configuration: Specify the Weblogic domain `<RTDCLM_Dev_Domain>`, and the Hostname and port values that apply to your environment.
 - Test: Click Test Connection.
 - Finish: Click Finish.
5. In the Select Server window, select the newly created AdminServer connection, and click Next.
 6. In the Weblogic options window, leave the options as they are, and click Finish.
- You should see this in the deployment tab:
- Application Deployed Successfully.
 - Elapsed time for deployment: [...]
 - ---- Deployment finished. ---

[Next time you can deploy the application by choosing the menu item Application -> Deploy -> `clm_application1` to AdminServer.]

The Oracle RTD Decision Management application is now deployed and accessible at the URL: `http://<server>:<port>/dm`

1.3 Preparing Your Development Application for Production

After you have finished configuring Oracle RTD Decision Management for your needs, you are ready to generate the `clm.ear` file to provide it to the domain administrator so it can be installed in the production environment following the instructions described in [Section 1.4, "Installing Oracle RTD Decision Management for Production."](#)

To generate this file, first review the Oracle RTD Decision Management configuration:

- In `clm\Build\metadata\<metadata_module>\security.xml` (and any other xml files in the same directory `clm\Build\metadata\<metadata_module>`), remove any `<users>` or `<enterprise-roles>` as the domain administrator will map Oracle RTD Decision Management application roles to your enterprise users and groups using Enterprise Manager (see [Section 1.5.1.2, "Adding Users in Oracle RTD Decision Management Metadata and Mapping Them to Decision Management Application Roles"](#)).
- Review `clm\Build\metadata\<metadata_module>\config\config.xml`. You will probably have to change the `workbenchService url` port from the development-site port number to the port where your production managed server will be running (for example, from the development-site Administration Server port 7001 to the production-site Managed Server port 7003).

If you have made changes to the configuration, go to the `clm\Build` directory and run the command **ant generate**:

Then run the command **ant ear**:

The `clm.ear` file is now located in the folder `clm\deploy` and is ready to be used in production.

You need to provide `load_core.sql` and `load_ils.sql` to set up the production database. These are located in folder `clm\Database\sql` and `clm\Database\sql\ils` respectively.

You also need to provide the Inline Service. It should be located in the folder `clm\Build\metadata\<metadata_module>\service`.

1.4 Installing Oracle RTD Decision Management for Production

This section contains the following topics:

- [Section 1.4.1, "Preparing for Installation"](#)
- [Section 1.4.2, "Production WebLogic Domain Creation"](#)
- [Section 1.4.3, "Installing Oracle RTD Decision Management"](#)

1.4.1 Preparing for Installation

The steps to perform in preparing to install Oracle RTD Decision Management are as follows:

1. Install Oracle Database.
2. Install Oracle WebLogic Server 11gR1.
Oracle Coherence and Oracle Enterprise Pack for Eclipse do not need to be installed and can be deselected during install.
3. Install Application Development Runtime.
4. Install Oracle RTD for WebLogic following chapter 2 of *Oracle Real-Time Decisions Installation and Administration Guide*.

Note: In addition to the instructions in the specified chapter, you must make the following extra provision for the Oracle RTD database SDDB:

- Before you can initialize the database either by using SDDBTool or by running the database initialization command line scripts, you must have the system privilege CREATE VIEW.
-

5. Create a WebLogic domain with a Managed Server, as explained in [Section 1.4.2, "Production WebLogic Domain Creation."](#)
6. Modify `nodemanager.properties` in `<middleware_home>/wlserver_10.3/common/nodemanager` and set **StartScriptEnabled** and **StopScriptEnabled** to true (you have to start Node Manager once for this file to exist).
7. Configure Oracle RTD to run on the Managed Server following chapter 5 (but not sections 5.1, 5.3, nor 5.5) of *Oracle Real-Time Decisions Installation and Administration Guide*.

The rest of this section contains the following topics:

- [Section 1.4.2, "Production WebLogic Domain Creation"](#)

1.4.2 Production WebLogic Domain Creation

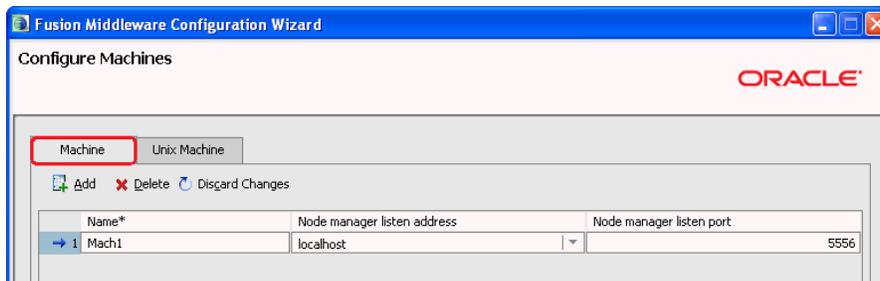
To create a production WebLogic domain, perform the following steps:

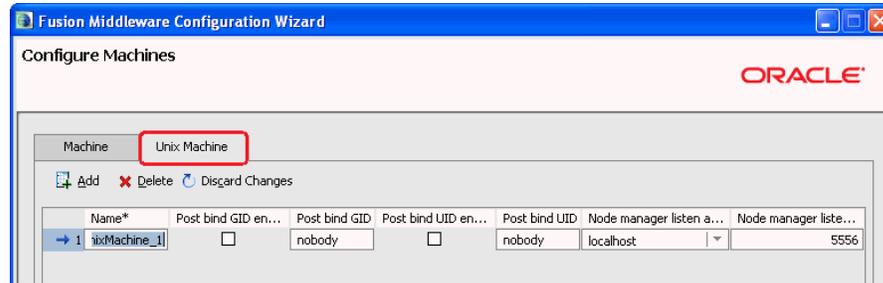
1. Start the Fusion Middleware Configuration Wizard.
On Windows, this is available via Start Menu: Programs -> Oracle Application Developer 11g -> Configure Application Server.
2. In the Welcome window, select **Create a new WebLogic domain**, then click Next.
3. In the Select Domain Source window:

- Select **Generate a domain configured automatically to support the following products.**
 - Select the **Oracle Enterprise Manager - 11.1.1.0 [oracle_common]** and **Oracle JRF - 11.1.1.0 [oracle_common]** options.
 - Click Next.
4. In the Specify Domain Name and Location, specify a name and location for your domain, then click Next.

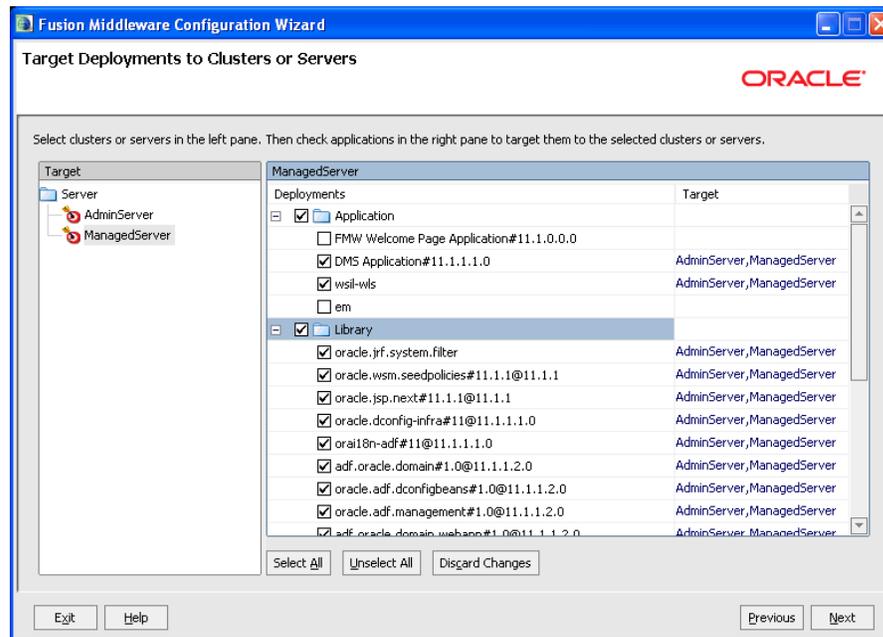
Note: You can specify any name for the domain. The generic name *<RTDCLM_Prod_Domain>* will be used to refer to the production environment domain name in all the setup steps of this and other sections.

5. In the Configure Administrator User Name and Password window, specify a user name and password for the administrator account, then click Next.
6. In the Configure Server Start Mode and JDK window, select **Production Mode**, then click Next.
7. In the Select Optional Configuration window, select the following options, then click Next.
- **Administration Server**
 - **Managed Servers, Clusters and Machines**
 - **Deployment and Services**
8. In the Configure the Administration Server window, optionally change the values as required for your production environment, then click Next
9. In the Configure Managed Servers window, add a managed server specifying a name of your choice, with address and port information appropriate to your environment, then click Next.
10. In the Configure Clusters window, click Next.
11. In the Configure Machines window:
- Select the Machine tab (for Windows) or the UNIX Machine tab (for UNIX)

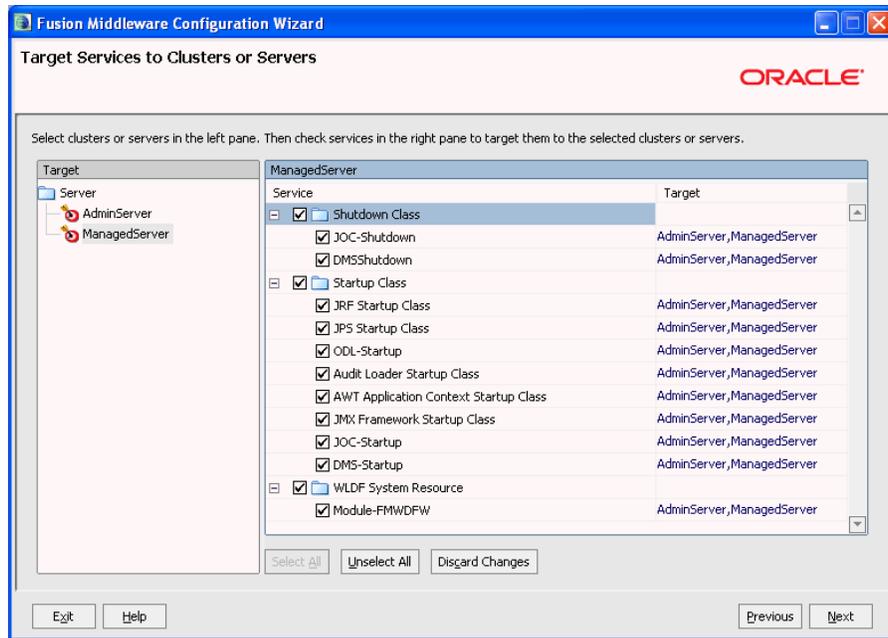




- Add a machine specifying a name of your choice, and with address and port information appropriate to your environment
Note the extra "Post bind" fields on the Unix Machine tab. Click Help in the Configure Machines window for further information on these fields, to determine their relevance for your environment.
 - Click Next
12. In the Assign Servers to Machines window, assign your managed server to the machine that you just specified, then click Next.
13. In the Target Deployments to Clusters or Servers window:
- Select the managed server as the Target
 - Select the applications **DMS Application#11.1.1.1.0** and **wsil-wls**
 - Select **Library** (which automatically selects all the libraries)



- Click Next
14. In the Target Services to Clusters or Servers window:
- Specify that all the services should be targeted to the managed server.



- Click Next.

15. Review the domain configuration details in the Configuration Summary window, then click Create.

Note: If you have performed these domain creation steps in sequence from [Section 1.4.1, "Preparing for Installation,"](#) continue those steps at step 6.

Other Configurations

For your own installation, you may have a different configuration based on your topology. For example:

- You can have the Administration Server on a different machine
- You can have multiple Managed Servers
- You can use a cluster
- You can enable SSL (see [Section 1.5.4, "Using SSL with Oracle RTD Decision Management Applications"](#))

1.4.3 Installing Oracle RTD Decision Management

After you have created the WebLogic domain and configured Oracle RTD to run on the Managed Server, you are ready to install Oracle RTD Decision Management.

This section contains the following topics:

- [Section 1.4.3.1, "Files Required from Development Environment"](#)
- [Section 1.4.3.2, "Oracle RTD Decision Management Database Creation"](#)
- [Section 1.4.3.3, "Oracle RTD Decision Management Data Source Setup"](#)
- [Section 1.4.3.4, "Oracle RTD Decision Management Data Source Access by Oracle RTD Setup"](#)

- [Section 1.4.3.5, "Storing Credentials to Make Oracle RTD Web Service Calls from Oracle RTD Decision Management"](#)
- [Section 1.4.3.6, "Oracle RTD Decision Management Application Deployment"](#)
- [Section 1.4.3.7, "Inline Service Deployment"](#)

1.4.3.1 Files Required from Development Environment

The Oracle RTD Decision Management configuration performed in the development environment is the source of the following files, required for the production environment:

- `clm.ear`
- `load core.sql`
- `load ils.sql`
- The Inline Service that is part of this application (see [Section 1.2.2.5, "Inline Service Deployment"](#))

Note: The `.sql` files are generated when you run `ant generate` (and before you run `ant ear`).

1.4.3.2 Oracle RTD Decision Management Database Creation

Note: For running the SQL commands and SQL scripts specified in this section, use SQL*Plus or an equivalent tool that can run on your database.

To create an Oracle RTD Decision Management database, perform the following steps:

1. Create a database user for the Oracle RTD Decision Management schema.

```
CREATE USER <username> IDENTIFIED BY <password> DEFAULT
TABLESPACE users TEMPORARY TABLESPACE temp QUOTA UNLIMITED ON
users;
```

```
GRANT CREATE VIEW, CONNECT, RESOURCE TO <username>;
```

Note: If you are going to use non 7-bit ASCII characters in choice group id or choice attribute ids, you must set the environment variable `NLS_LANG` to `.UTF8` before running the `sql` scripts in the steps that follow.

2. Create the schema, by performing the following steps:

- Go to the subfolder `clm\Database\sql` of where you unzipped `RTD_Designer.zip`, then run the file `load core.sql`.

Note: The file `load core.sql` contains generic commands, required for all Oracle RTD Decision Management applications. The commands in `load ils.sql`, as released, are for the RTD for Marketing Optimization application.

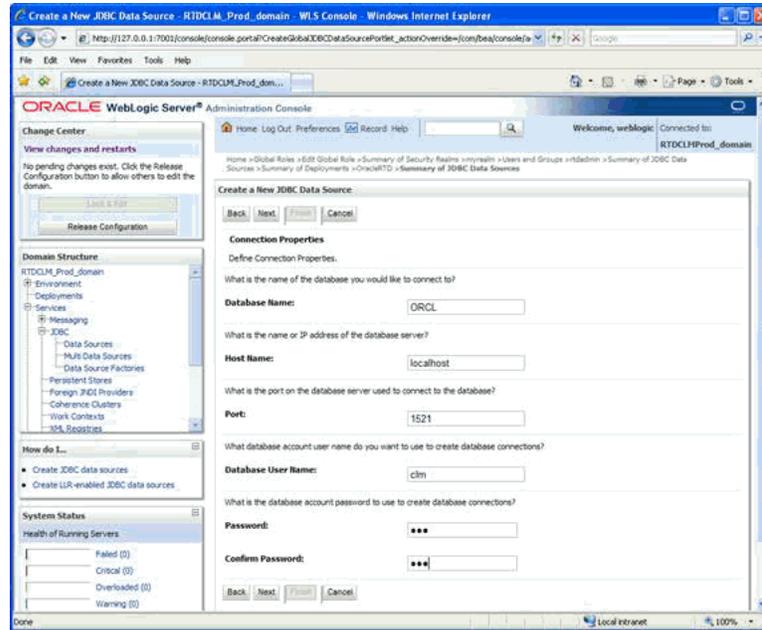
- Go to the subfolder `clm\Database\sql\ils` of where you unzipped `RTD_Designer.zip`, then run the file `load ils.sql`.

Make sure you have committed your changes.

1.4.3.3 Oracle RTD Decision Management Data Source Setup

To set up an Oracle RTD Decision Management data source, perform the following steps:

1. Start the Administration Server.
On Windows, this is available via the Start Menu: Programs -> Oracle WebLogic -> User Projects -> `<RTDCLM_Prod_Domain>` -> Start Admin Server for Weblogic Server Domain
2. Enter the administrator username and password, which was specified during domain creation (see [Section 1.2.1.3, "Creating the WebLogic Domain"](#)).
3. Open the Admin Server Console.
On Windows, this is available via the Start Menu: Programs -> Oracle WebLogic -> User Projects -> `<RTDCLM_Prod_Domain>` -> Admin Server Console.
4. Log in with the administrator username and password.
5. Click **Lock & Edit** to make change.
6. Navigate the path Services -> Data Sources -> New -> Generic Data Source, then enter:
 - Name: **CLM_DS**
 - JNDI Name: **CLMDS**
 - Database Type: **Oracle**Click Next.
7. For the Database Driver, select **Oracle's Driver (Thin) for Instance connections; Versions: 9.0.1 and later**, then click Next.
8. In the Transaction Options window, deselect **Supports Global Transactions**, then click Next.
9. For the Connection Properties:
 - Specify the database settings that match your configuration, for example:



Ensure that the Database User name and Password match the values that you set up in step 1 of [Section 1.4.3.2, "Oracle RTD Decision Management Database Creation."](#)

- Click Next.

10. Leave all the settings already filled, except enter CHOICE for the Test Table Name, and click Test Configuration. Then click Next.
11. Select the manager server that you specified in [Section 1.4.2, "Production WebLogic Domain Creation"](#) as the target, then click Finish.
12. Select **Activate Changes**.

1.4.3.4 Oracle RTD Decision Management Data Source Access by Oracle RTD Setup

Section 8.3.3 of *Oracle Real-Time Decisions Installation and Administration Guide* describes the general steps to add a data source to Oracle RTD. Follow the steps in that section, and, specifically for the Oracle RTD Decision Management data source, add the following in `web.xml` for `rtis.war:8`

```
<resource-ref id="CLM_RTIS">
  <res-ref-name>CLMDS</res-ref-name>
  <res-type>javax.sql.DataSource</res-type>
  <res-auth>Container</res-auth>
  <res-sharing-scope>Unshareable</res-sharing-scope>
</resource-ref>
```

and add the following in `web.xml` for `soap.war:`

```
<resource-ref id="CLM_Axis">
  <res-ref-name>CLMDS</res-ref-name>
  <res-type>javax.sql.DataSource</res-type>
  <res-auth>Container</res-auth>
  <res-sharing-scope>Unshareable</res-sharing-scope>
</resource-ref>
```

1.4.3.5 Storing Credentials to Make Oracle RTD Web Service Calls from Oracle RTD Decision Management

Oracle RTD Decision Management makes web service calls from the Decision Management server side to Oracle RTD. Credentials for these calls are stored in the WebLogic credential store. In summary form, the credential setup process is as follows:

- First, you create a user that will be used to authenticate the web service calls
- Then, you store the credentials of this user in the WebLogic credential store, so that the Decision Management application can retrieve them

To create the user and to store the user credentials in the WebLogic credential store, perform the following steps:

1. Open the Admin Server Console.
On Windows, this is available via Start Menu: Programs -> Oracle WebLogic -> User Projects -> <RTDCLM_Prod_Domain> -> Admin Server Console.
2. Log in with the administrator username and password.
3. Select Security Realms and select the security realm you are using (myrealm by default).
4. Select the Users and Groups tab. and click New to create a new user.
5. Specify a username and password (generic names <clmclient_user> and <clmclient_password>).
6. Click OK.
7. Select the user that you just created and select the Groups tab.
8. Select the RTDDCUserGroup group, shuttle it to the right.
9. Click Save.
10. Open the Enterprise Manager on the Administration Server.
11. Log in with the administrator username and password.
12. In the Target Navigation Pane, select WebLogic Domain -> <RTDCLM_Prod_Domain>.
13. In the <RTDCLM_Prod_domain> window, from the WebLogic Domain dropdown menu, select Security -> Credentials.
14. In the Credentials area, click Create Map, and in the Create Map window, enter **oracle.rtd.clm** and click OK.
15. Click Create Key, and in the Create Key window:
 - Enter the Key **clm.client** for the map that you just created
 - Enter the same User Name and Password as you created previously in this process
 - Click OK
16. In the Create Map window, enter oracle.rtd.clm, then click OK.

This completes the set up of the credentials for accessing Oracle RTD from Oracle RTD Decision Management.

1.4.3.6 Oracle RTD Decision Management Application Deployment

The final step is to deploy the Oracle RTD Decision Management application. You should have a `clm.ear` file that was generated from the development environment. The `clm.ear` application will be deployed to the production environment.

Note: Enterprise Manager must be used to deploy the Oracle RTD Decision Management application, Weblogic Admin console cannot be used.

1. Start Node Manager.

On Windows, Node Manager is either running as a service, or you can start it from the Start Menu: Programs -> Oracle WebLogic -> WebLogic 11gR1 -> Tools -> Node Manager.

2. Start Node Manager. On Windows, it is either running as a service, or you can start it from the Start Menu: Programs -> Oracle WebLogic -> WebLogic 11gR1 -> Tools -> Node Manager.
3. Open the Enterprise Manager on the Administration Server.
4. Log in with the administrator username and password.
5. In the Target Navigation Pane, select WebLogic Domain -> `<RTDCLM_Prod_Domain>`.
6. In the `<RTDCLM_Prod_Domain>` window, from the WebLogic Domain dropdown menu, select Control -> Start Up.
7. In the `<RTDCLM_Prod_Domain>` window, from the WebLogic Domain dropdown menu, select Application Deployment -> Deploy...
8. In the Select Archive window, select the `.ear` file that was originally generated in the development environment.
9. In the Select Target window, select the managed server only.
10. In the Application Attributes window, do not alter any values.
11. In the Deployment Settings window, click Configure Application Security.
12. In the Configure Application Security, for your first deployment, select Append for both the Application Policy Migration and the Application Credential Migration, and click Apply.

Note: Use the Append options for first-time deployment only. For subsequent deployments:

- For Application Policy Migration, select Overwrite.
 - The Application Credential Migration does not appear.
-
-

13. Back in the Deployment Settings window, click Deploy.

The Oracle RTD Decision Management application is now deployed and accessible at the URL: `http://<server>:<port>/dm`.

1.4.3.7 Inline Service Deployment

The application that you deployed requires a specific Inline Service to be running in the Oracle RTD instance on the same server as Oracle RTD Decision Management.

You should have received this Inline Service with the `clm.ear` file. Using Decision Studio, you must deploy that Inline Service to the Oracle RTD instance running on the same server as Oracle RTD Decision Management.

1.5 Security Configuration

There are three ways to setup authentication for Oracle RTD Decision Management:

- In the development environment, you can manage the users within JDeveloper
- In the development environment, you can map your enterprise users and groups to Decision Management application roles using WebLogic Admin Console
- In the production environment, you map your enterprise users and groups to Decision Management application roles using Enterprise Manager.

This section contains the following topics:

- [Section 1.5.1, "Managing Users Within JDeveloper in a Development Environment"](#)
- [Section 1.5.2, "Mapping Enterprise Users and Groups to Decision Management Application Roles in a Development Environment"](#)
- [Section 1.5.3, "Mapping Enterprise Users and Groups to Decision Management Application Roles in a Production Environment"](#)
- [Section 1.5.4, "Using SSL with Oracle RTD Decision Management Applications"](#)

1.5.1 Managing Users Within JDeveloper in a Development Environment

The simplest way to configure access to a Decision Management application in development mode is to specify users in Oracle RTD Decision Management metadata and map them to Decision Management application roles in Oracle RTD Decision Management metadata. When deploying from JDeveloper, this will automatically create the users in WebLogic and map them to the Decision Management application roles.

This is done with these steps:

- Retrieve the encrypted password for your user
- Add the user in Oracle RTD Decision Management metadata and map the user to Decision Management application roles
- Run application generation
- Deploy the Decision Management application
- Add the user to Oracle RTD groups

This section contains the following topics:

- [Section 1.5.1.1, "Retrieving the Encrypted Password for Your User"](#)
- [Section 1.5.1.2, "Adding Users in Oracle RTD Decision Management Metadata and Mapping Them to Decision Management Application Roles"](#)
- [Section 1.5.1.3, "Running Application Generation"](#)
- [Section 1.5.1.4, "Deploying the Oracle RTD Decision Management Application"](#)

- [Section 1.5.1.5, "Adding Users to Oracle RTD Groups"](#)

1.5.1.1 Retrieving the Encrypted Password for Your User

Passwords are stored encrypted in `jazn-data.xml`. You will need to access the encrypted password and add the encrypted password explicitly to Oracle RTD Decision Management metadata in a subsequent setup step.

You can use JDeveloper to get the encrypted password string, as follows:

1. Open JDeveloper from Start Menu, Programs -> Oracle Fusion Middleware <11g_version> -> JDeveloper Studio <11g_version>.
2. Select Default Role.
3. Click Open Application... and select **clm.jws** in the `clm` subfolder of where you unzipped `RTD_Designer.zip`.
4. In Application Navigator, open Application Resources.
5. Expand Descriptors -> META-INF, and double-click **jazn-data.xml**.
6. In the Users tab, create a user by clicking on the + icon next to Users.
7. Enter a user Name and a Password.
8. Click the Source tab (from the tab list at the bottom of the window).

In the source XML, you can now locate the credentials for the user that you just created. The credentials value is the encrypted password for the user.

For example, after entering user name `clmuser` and a password for `clmuser`, this could appear in the Source (*with the encrypted password highlighted in bold in the example below*) as:

```
<user>
  <name>clmuser</name>
  <credentials>{903}1oyIIwIDNKqdw4D9XrMhQZ4yEFLXim+</credentials>
</user>
```

9. Copy the encrypted password into your standard machine buffer (Ctrl-C). Retain it there for use in the subsequent setup step, "[Adding Users in Oracle RTD Decision Management Metadata and Mapping Them to Decision Management Application Roles](#)".

1.5.1.2 Adding Users in Oracle RTD Decision Management Metadata and Mapping Them to Decision Management Application Roles

The description in this section assumes that you are editing the Decision Manager for Marketing Optimization implementation. If you use a different application, see [Chapter 2, "Configuring Oracle RTD Decision Management"](#) for the differences.

To add users in Oracle RTD Decision Management metadata and to map them to Decision Management application roles, perform the following steps:

1. Go to the `clm/Build/metadata/ref` folder.
2. Edit `security.xml`, or create a different file, such as `users.xml`.

Add user information (including the encrypted password that you saved in the previous setup step) using the following as an example (this example assigns the **CLMAdministrator** application role to the user **clmuser**):

```
<users>
  <user>
```

```
<name>clmuser</name>
<display-name>clmuser</display-name>
<description>this is a clm user</description>
<credentials>{903}1oyIIwIDNKqdw4D9XrMhQDZ4yEFLXim+</credentials>
<application-roles>CLMAdministrator</application-roles>
</user>
</users>
```

1.5.1.3 Running Application Generation

To run application generation, perform the following steps

1. Open a command prompt in the clm/Build directory.
2. Run the command **ant generate -Dchanges=overwrite**.

Note: You must overwrite changes because you made changes to `jazn-data.xml` in the previous setup step when you manually entered a password in JDeveloper, and here the ant generation overwrites this file.

In JDeveloper, verify that the user now shows up in `jazn-data.xml` and is mapped to the CLMAdministrator application role.

1.5.1.4 Deploying the Oracle RTD Decision Management Application

In JDeveloper, deploy the Decision Management application to AdminServer by choosing menu item: Application -> Deploy -> clm_application1.

1.5.1.5 Adding Users to Oracle RTD Groups

After the user has been created in WebLogic, you must add the user to the RTDDCUserGroup and the RTDChoiceEditorGroup groups, by performing the following steps:

1. Open the Admin Server Console.
On Windows, this is available via the Start Menu: Programs -> Oracle Fusion Middleware <11g_version> -> User Projects -> <RTDCLM_Dev_Domain> -> Admin Server Console.
2. Log in with the administrator username and password.
3. Select Security Realms and select the security realm you are using (myrealm by default).
4. Select the Users and Groups tab.
5. Click the user that you created in [Section 1.5.1.1, "Retrieving the Encrypted Password for Your User."](#)
6. Select the Groups tab.
7. Select RTDDCUserGroup and RTDChoiceEditorGroup, and shuttle them to the right.
8. Click Save.

1.5.2 Mapping Enterprise Users and Groups to Decision Management Application Roles in a Development Environment

If you want to use your enterprise users and groups in development mode, you specify enterprise roles in Decision Management metadata and map them to Decision Management application roles. When deploying from JDeveloper, this will automatically create groups in WebLogic for these enterprise roles and map them to the Decision Management application roles.

This is done with these steps:

- Add the enterprise role in Decision Management metadata and map it to Decision Management application roles
- Run application generation
- Deploy the Oracle RTD Decision Management application
- Specify this group as a member of Oracle RTD groups
- Map your enterprise users or groups to this Decision Management group

This section contains the following topics:

- [Section 1.5.2.1, "Adding the Enterprise Role in Oracle RTD Decision Management Metadata and Mapping it to Decision Management Application Roles"](#)
- [Section 1.5.2.2, "Running Decision Management Application Generation"](#)
- [Section 1.5.2.3, "Deploying the Decision Management Application"](#)
- [Section 1.5.2.4, "Specify the WebLogic Group as a Member of Oracle RTD Groups"](#)
- [Section 1.5.2.5, "Mapping Your Enterprise Users or Groups to the Oracle RTD Decision Management Group"](#)

1.5.2.1 Adding the Enterprise Role in Oracle RTD Decision Management Metadata and Mapping it to Decision Management Application Roles

The description in this section assumes that you are editing the RTD for Marketing Optimization implementation. If you use a different application, see [Chapter 2, "Configuring Oracle RTD Decision Management"](#) for the differences.

To add the enterprise role in Oracle RTD Decision Management metadata and to map it to Decision Management application roles, perform the following steps:

1. Go to the `clm/Build/metadata/ref` folder.
2. Edit `security.xml`, or create a different file, such as `groups.xml`.

Add the following (here we create a **clmgroup** enterprise role and map it to the **CLMAdministrator** application role):

```
<enterprise-roles>
  <enterprise-role>
    <name>clmgroup</name>
    <application-roles>CLMAdministrator</application-roles>
  </enterprise-role>
</enterprise-roles>
```

1.5.2.2 Running Decision Management Application Generation

To run application generation, perform the following steps

1. Open a command prompt in the `clm/Build` directory.

2. Run the command **ant generate**.

In JDeveloper, verify that the group now shows up in `jazn-data.xml` and is mapped to the CLMAdministrator application role.

1.5.2.3 Deploying the Decision Management Application

In JDeveloper, deploy the Decision Management application to AdminServer by choosing the menu item: Application -> Deploy -> `clm_application1`.

1.5.2.4 Specify the WebLogic Group as a Member of Oracle RTD Groups

After the enterprise role has been created as a group in WebLogic, you must make it a member of the RTDDCUserGroup and the RTDChoiceEditorGroup groups, by performing the following steps:

1. Open the Admin Server Console.
On Windows, this is available via the Start Menu: Programs -> Oracle Fusion Middleware `<11g_version>` -> User Projects -> `<RTDCLM_Dev_Domain>` -> Admin Server Console.
2. Log in with the administrator username and password.
3. Select Security Realms and select the security realm you are using (myrealm by default).
4. Select the Users and Groups tab.
5. Select the Groups tab.
6. Click **clmgroup**.
7. Click the Membership tab.
8. Select RTDDCUserGroup and RTDChoiceEditorGroup, and shuttle them to the right.
9. Click Save.

1.5.2.5 Mapping Your Enterprise Users or Groups to the Oracle RTD Decision Management Group

You can now grant access to Oracle RTD Decision Management to your users by adding them the **clmgroup** group.

You can also grant access to Oracle RTD Decision Management to your groups by making them a member of the **clmgroup** group.

1.5.3 Mapping Enterprise Users and Groups to Decision Management Application Roles in a Production Environment

In a production environment, you typically manage your users and groups outside of Oracle RTD Decision Management, and only map these users and groups to Decision Management application roles.

In this section, we will use "clmuser" and "clmgroup" as examples of your enterprise users and groups.

This section contains the following topics:

- [Section 1.5.3.1, "Mapping Users or Groups to Decision Management Application Roles"](#)

- [Section 1.5.3.2, "Adding the User or Group to Oracle RTD Groups"](#)

1.5.3.1 Mapping Users or Groups to Decision Management Application Roles

To map users or groups to Decision Management application roles, perform the following steps:

1. Open the Enterprise Manager on the Administration Server.
2. Log in with the administrator username and password.
3. In the Target Navigation Pane, select the clm deployment: Application Deployments -> Internal Applications -> clm.
4. In the clm window, from the Application Deployment dropdown menu, select Security -> Application Roles and click the "Search application roles" button.
5. Click CLMAdministrator.
6. To add a user, click on Add User, search for your user and shuttle it to the right.
7. Click OK.
8. To add a group, click on Add Group, search for your group and shuttle it to the right.
9. Click OK twice.

1.5.3.2 Adding the User or Group to Oracle RTD Groups

To use Oracle RTD Decision Management, your user or groups must be in the RTDDCUserGroup and RTDChoiceEditorGroup groups.

To add users or groups to the Oracle RTD groups, perform the following steps:

1. Open the Admin Server Console.
On Windows, this is available via Start Menu: Programs -> Oracle WebLogic -> User Projects -> <RTDCLM_Prod_Domain> -> Admin Server Console.
2. Log in with the administrator username and password.
3. Select Security Realms and select the security realm you are using (myrealm by default).
4. Select the Users and Groups tab.
5. Click **clmuser**.
6. Select the Groups tab.
7. Select RTDDCUserGroup and RTDChoiceEditorGroup, and shuttle them to the right.
8. Go back to the Users and Groups tab.
9. Select the Groups tab.
10. Click **clmgroup**, and click the Membership tab.
11. Select RTDDCUserGroup and RTDChoiceEditorGroup, and shuttle them to the right.
12. Click Save.

1.5.4 Using SSL with Oracle RTD Decision Management Applications

This section contains the following topics:

- [Section 1.5.4.1, "SSL for Development Environment"](#)
- [Section 1.5.4.2, "SSL for Production Environment"](#)

1.5.4.1 SSL for Development Environment

To set up SSL for the development environment, perform the following steps:

1. Configure keystore and trust store as described in Chapter 2.6, "Using SSL with Oracle Real-Time Decisions" in *Oracle Real-Time Decisions Installation and Administration Guide*.
2. Perform the instructions in Section 5.13, "Configuring SSL for Real-Time Decision Server" of *Oracle Real-Time Decisions Installation and Administration Guide*.

The remainder of this sequence assumes that you have successfully installed the Oracle RTD certificates and can access the non-SSL ports, but does not assume that you have disabled the SSL ports.

Please note that the Oracle RTD certificates are for testing only and should not be used in production.

3. Open the Admin Server Console.

On Windows, this is available via the Start Menu: Programs -> Oracle Fusion Middleware <11g_version> -> User Projects -> <RTDCLM_Dev_domain> -> Admin Server Console.

4. Log in with the administrator username and password.
5. Navigate the path Environments > Servers > AdminServer.
6. Select AdminServer, then Configuration tab -> SSL tab > Advanced.
7. Select "Use Server Certs". Save your changes to the SSL tab.
8. Add the following to <domain> \bin\setDomainEnv.cmd:

```
-Djavax.net.ssl.trustStore=<RTD_HOME>\RTD\etc\ssl\sdtrust.store  
or (Unix)
```

```
-Djavax.net.ssl.trustStore=<RTD_HOME>/RTD/etc/ssl/sdtrust.store
```

9. Change clm\Build\metadata\<metadata_module>\config\config.xml workbenchClient to https://<server>:<SSL port>.
10. Run ant generate.
11. Stop and delete the existing Oracle RTD Decision Management application deployment.
12. Redeploy the application from JDeveloper using the connection you set up in [Section 1.2.2.7, "Oracle RTD Decision Management Application Deployment."](#)
13. Verify that both the Admin Server Console and the Oracle RTD Decision Management application are accessible using the SSL port. After you have determined this, you should consider disabling the non-SSL Admin Server port.

Ignore the certificate errors from the browser. Note again that the Oracle RTD test certificate is for testing only.
14. Stop and restart all services.
15. Access Oracle RTD Decision Management through the appropriate secure port, for example, <https://myserver:7002/dm>.

1.5.4.2 SSL for Production Environment

To set up SSL for the production environment, perform the following steps:

1. Configure keystore and trust store as described in Chapter 2.6, "Using SSL with Oracle Real-Time Decisions" in *Oracle Real-Time Decisions Installation and Administration Guide*.
2. Perform the instructions in Section 5.13, "Configuring SSL for Real-Time Decision Server" of *Oracle Real-Time Decisions Installation and Administration Guide*.

The remainder of this sequence assumes that you have successfully installed the Oracle RTD certificates and can access the non-SSL ports, but does not assume that you have disabled the SSL ports.

Please note that the Oracle RTD certificates are for testing only and should not be used in production.

3. Open the Admin Server Console.

On Windows, this is available via the Start Menu: Programs -> Oracle Fusion Middleware <11g_version> -> User Projects -><RTDCLM_Prod_domain> -> Admin Server Console.

4. Log in with the administrator username and password.
5. Navigate the path Environments > Servers > <Managed_Server>.
6. Select the Managed Server, then Configuration tab -> SSL tab > Advanced.
7. Select "Use Server Certs". Save your changes to the SSL tab.
8. Add the following to <domain> \bin\setDomainEnv.cmd:

```
-Djavax.net.ssl.trustStore=<RTD_HOME>\RTD\etc\ssl\sdtrust.store
```

or (Unix)

```
-Djavax.net.ssl.trustStore=<RTD_HOME>/RTD/etc/ssl/sdtrust.store
```

Alternatively, add this to the managed server "Server Start" -> "Arguments" box.

9. Change clm\Build\metadata\<metadata_module>\config\config.xml workbenchClient to https://<server>:<SSL_port>.
10. Run ant generate.
11. Stop and delete the existing Oracle RTD Decision Management application deployment.
12. Redeploy the application to non-SSL port. Do this by using Enterprise Manager deployed on Admin Server and listening non-SSL port.
13. Verify that the Oracle RTD Decision Management application is accessible using the SSL port.

Ignore the certificate errors from the browser. Note that the Oracle RTD test certificate is for testing only.
14. Stop and restart all services.
15. Access Oracle RTD Decision Management through the appropriate secure port, for example, <https://myserverx:7002/dm>.

Configuring Oracle RTD Decision Management

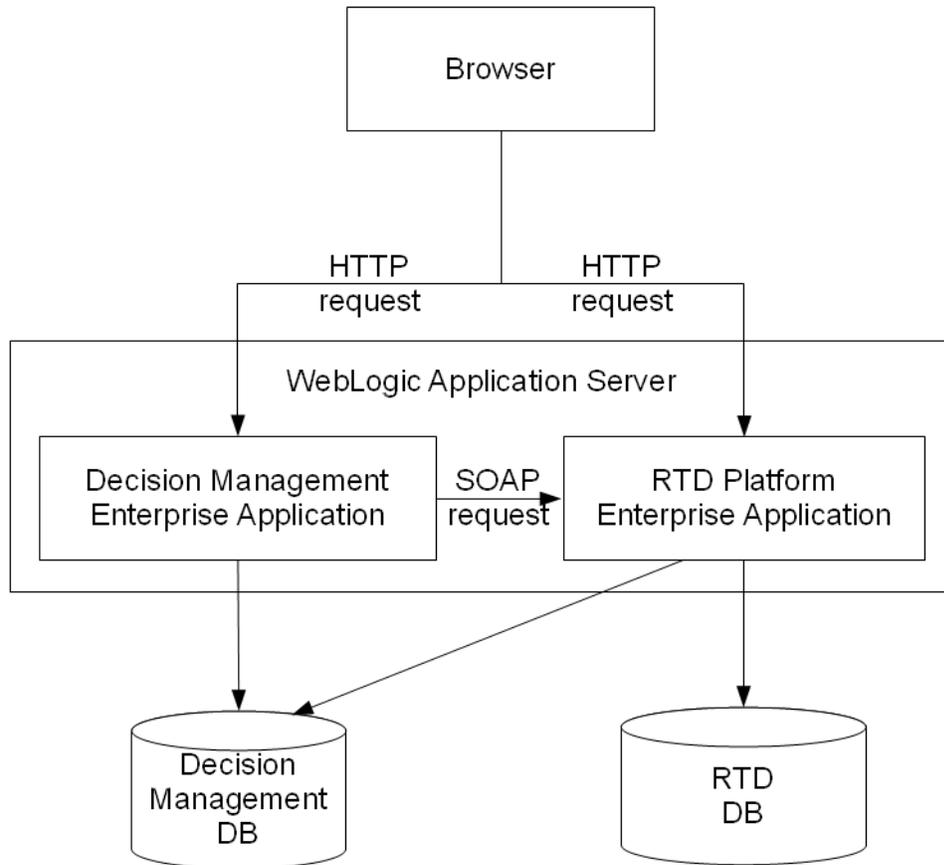
Terminology: The term "reference implementation" is used in this chapter to refer to the specific Oracle RTD Decision Management application RTD for Marketing Optimization, released with Oracle RTD Base Application.

This chapter contains the following topics:

- [Section 2.1, "Oracle RTD Decision Management Architecture Overview"](#)
- [Section 2.2, "Configuring Oracle RTD Decision Management"](#)
- [Section 2.3, "Miscellaneous"](#)

2.1 Oracle RTD Decision Management Architecture Overview

The following diagram shows the communications at runtime between the various components of Oracle RTD Decision Management after installation:



The Oracle RTD Decision Management application is an ADF based web application, which also leverages the Oracle RTD rule editor and reports as JSP pages embedded in an IFrame. As such, the Decision Manager browser displays JSP pages from both the Oracle RTD Decision Management and the Oracle RTD applications.

Users log in to the Oracle RTD Decision Management application and credentials are carried over to the Oracle RTD application because the Weblogic JSESSIONID cookie is shared between both applications (which must reside in the same application server).

Additionally, Oracle RTD makes web service (SOAP) calls from the Oracle RTD Decision Management application to the Oracle RTD workbench server to retrieve type restrictions. These requests use the username and password that you configured

in [Section 1.4.3.5, "Storing Credentials to Make Oracle RTD Web Service Calls from Oracle RTD Decision Management."](#)

The Oracle RTD Decision Management application accesses the Decision Management database using the credentials defined in the CLMDS JNDI datasource.

The Oracle RTD Decision Management Inline Service loaded in the Oracle RTD application accesses the Decision Management database to retrieve dynamic choices.

The Oracle RTD application accesses the Oracle RTD (SDDS) database using the credentials defined in the SDDS JNDI datasource.

2.2 Configuring Oracle RTD Decision Management

This section contains the following topics:

- [Section 2.2.1, "Overview"](#)
- [Section 2.2.2, "Oracle RTD Decision Management Metadata Configuration Files"](#)
- [Section 2.2.3, "Ant Tasks"](#)
- [Section 2.2.4, "Java API"](#)
- [Section 2.2.5, "Application Extensions"](#)

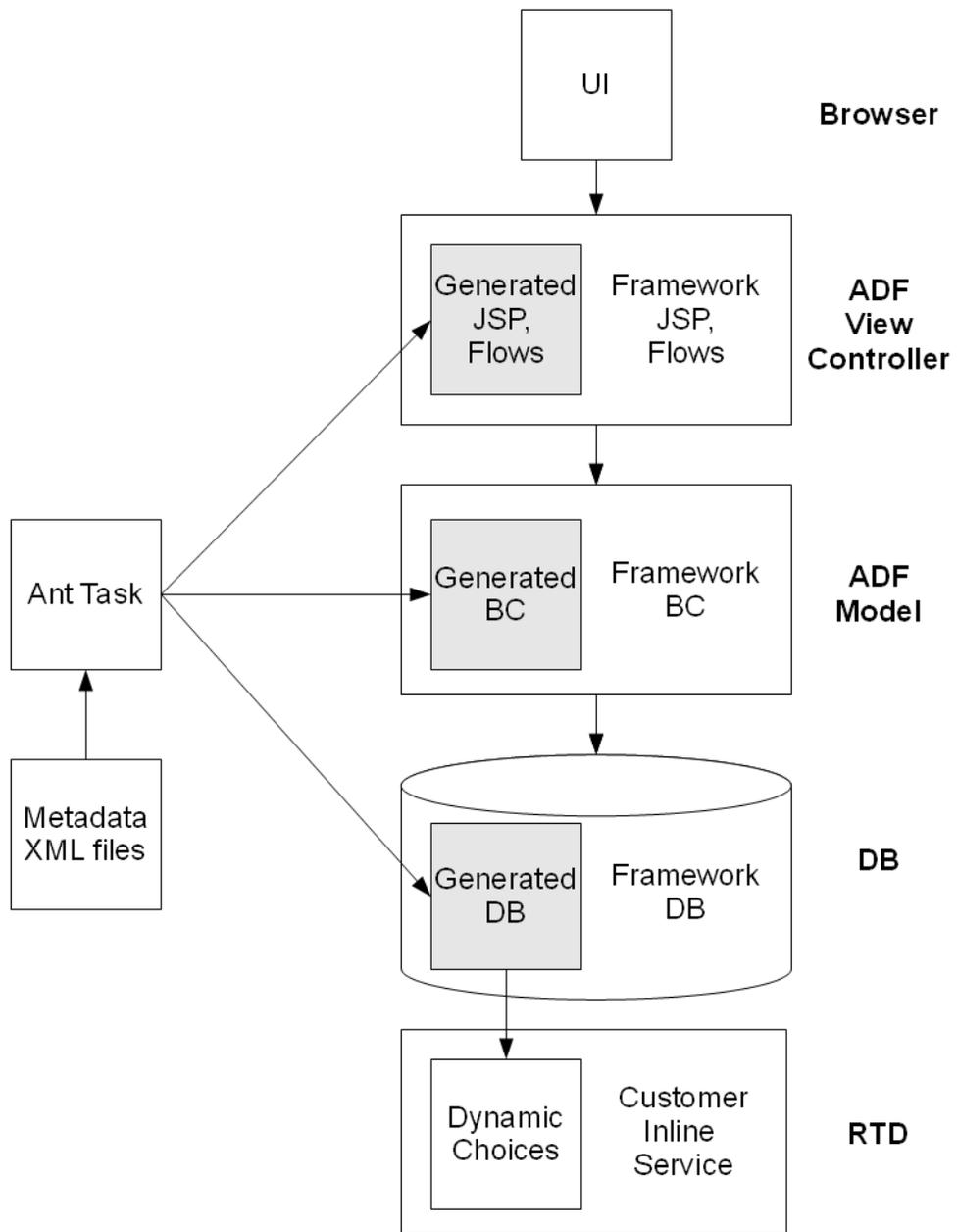
2.2.1 Overview

Oracle RTD Decision Management applications are built and deployed using Decision Designer.

Decision Designer is the combination of design tools that include JDeveloper, ant, and the metadata files that you set up and used to create the Oracle RTD Decision Management application, as described in [Chapter 1, "Installing Oracle RTD Decision Management."](#)

You can use Decision Designer to make changes to the reference implementation, or to build your own application from scratch.

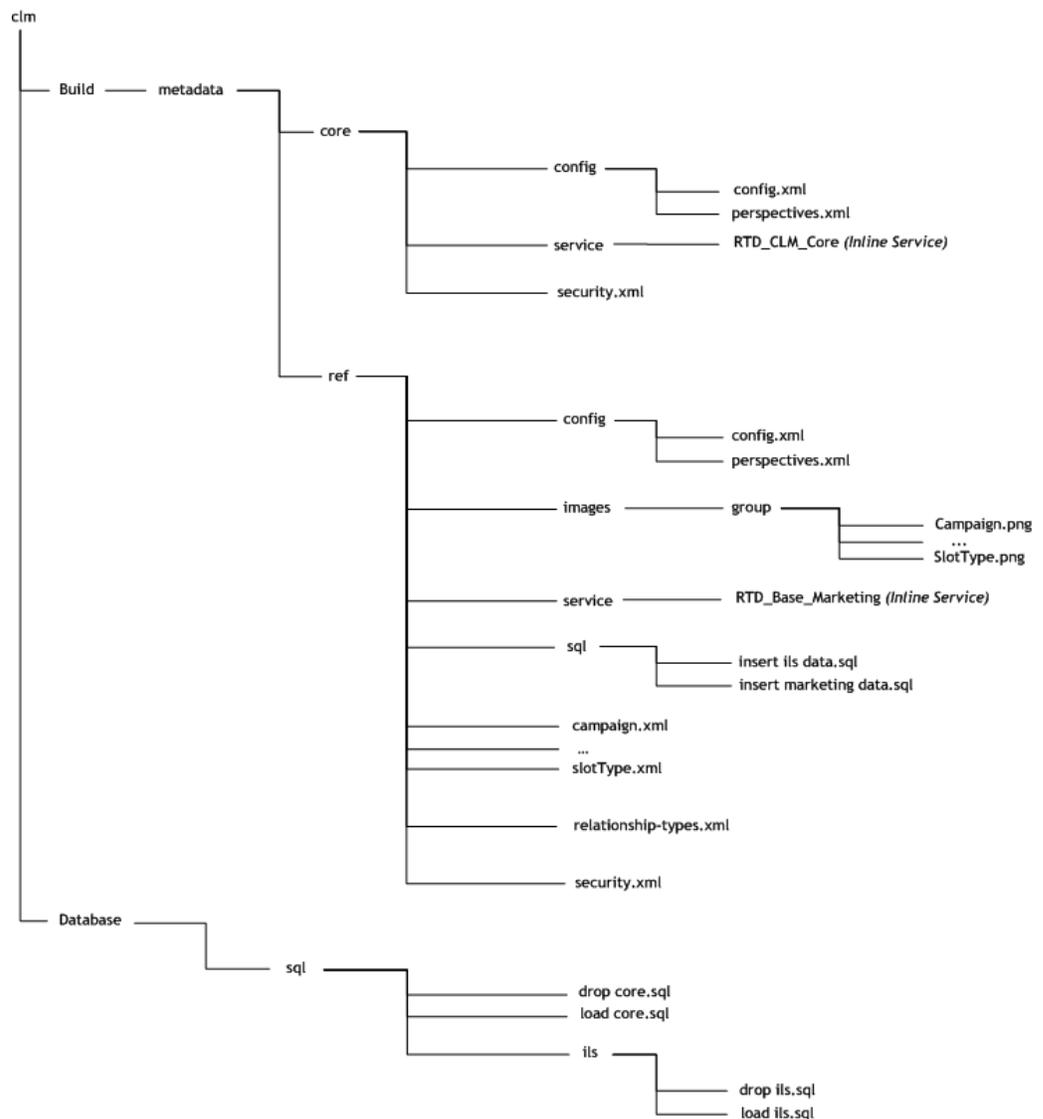
The following diagram shows what components are generated by Decision Designer:



2.2.2 Oracle RTD Decision Management Metadata Configuration Files

The following diagram shows the main Decision Management metadata and database setup files, as released with Oracle RTD Base Application.

Figure 2–1 Main Oracle RTD Decision Management Metadata and Database Files



The folder **clm\Build\metadata**, also referred to generically as *<metadata_modules_home>*, contains metadata modules for different applications.

The folder **clm\Build\metadata\ref** contains the metadata for the reference implementation.

Within a metadata module, two files, `config.xml` and `perspectives.xml`, must be in the **config** subdirectory. And you can add as many files as you want in the main directory to configure choice groups, relationship types, and security.

The folder **clm\Build\metadata\core** contains the metadata module for the core application. Use the core application when you want to create your own application from scratch, it contains the strict minimum for an empty application and Inline Service.

This section contains the following topics:

- [Section 2.2.2.1, "Config XML File"](#)
- [Section 2.2.2.2, "Perspectives XML File"](#)

- [Section 2.2.2.3, "Choice Group and Relationship-Types XML Files"](#)
- [Section 2.2.2.4, "Security XML Files"](#)
- [Section 2.2.2.5, "Folder for Choice Group Images"](#)
- [Section 2.2.2.6, "Inline Service Folder"](#)

2.2.2.1 Config XML File

`config.xml` contains general configuration settings:

Under a `<config>` node, the following settings are available:

- `<workbenchService>` specifies the url of the Oracle RTD server running in the same host WebLogic server as Oracle RTD Decision Management.

The host is therefore always **localhost**, but the port can change based on your configuration. If you followed the instructions in [Chapter 1, "Installing Oracle RTD Decision Management,"](#) the port is 7001 in a development environment and 7003 in a production environment.
- `<inlineService>` specifies the name of the Inline Service used to get type restrictions and decision center reports.

The deployment state must be the exact deployment state that you want to access.
- `<decisionCenter>` specifies the base URL to access embedded decision center reports. Remove the `<decisionCenter>` node to hide the decision center reports in the user interface.
- `<ownership-mode>` specifies the ownership mode for when a choice is edited in a project. The value can be:
 - off (the default)

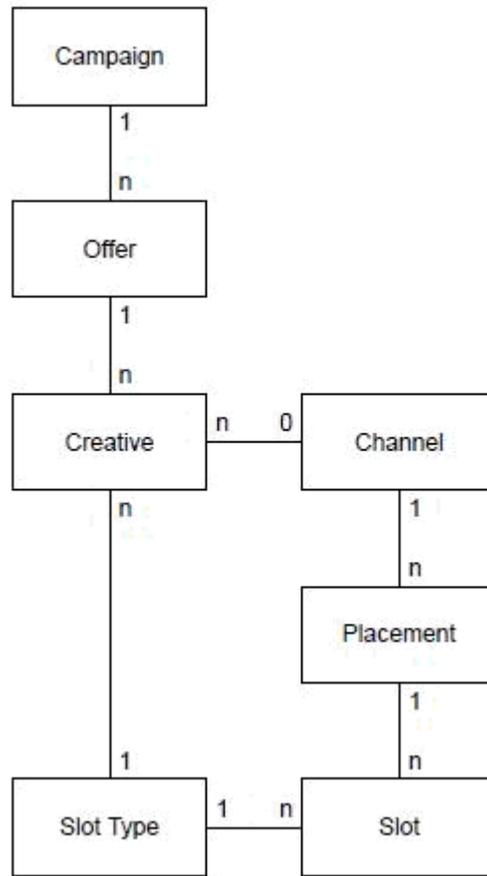
The user does not become the owner of the choice when the user edits a choice, and the choice owner is not shown in the Decision Manager user interface
 - on

The user becomes the owner of the choice when the user edits a choice, and the choice owner is shown in the Decision Manager user interface
- `<display-name>` specifies the name of the application as it appears on the login page and at the top of the main page

2.2.2.2 Perspectives XML File

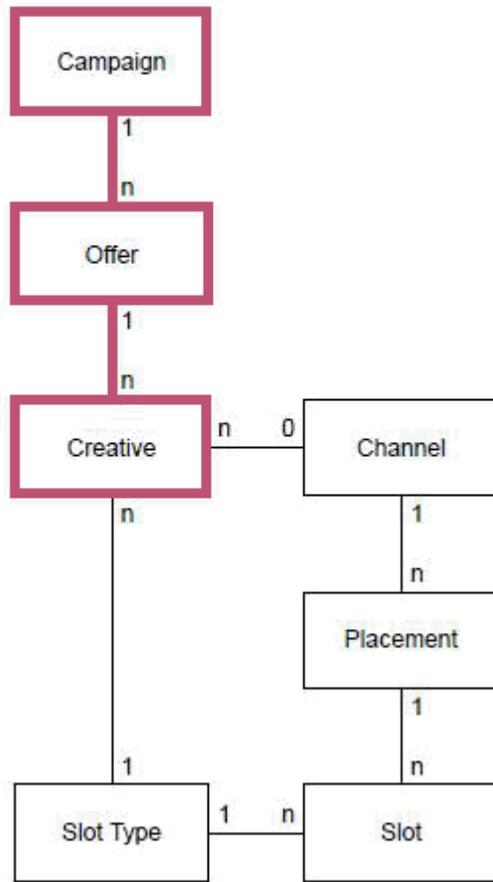
Perspectives allow business users to view a subset of the choice group graph in the tree in the Decision Manager user interface.

For instance, the reference implementation contains the choices shown in the following choice group graph:

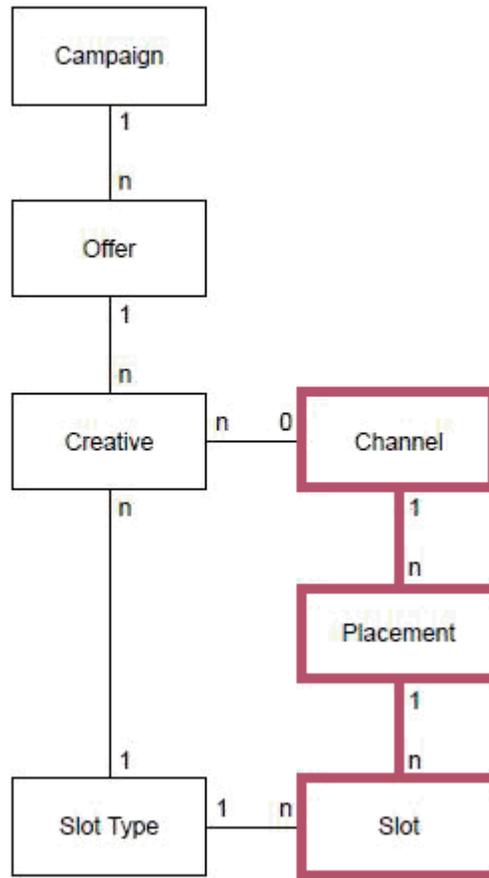


Four "general" - that is, unqualified - perspectives are defined on top of this graph:

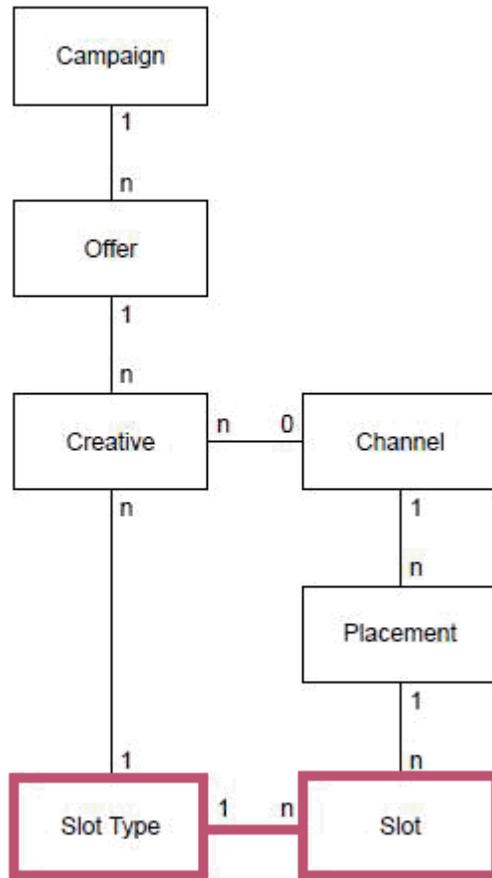
Campaigns perspective



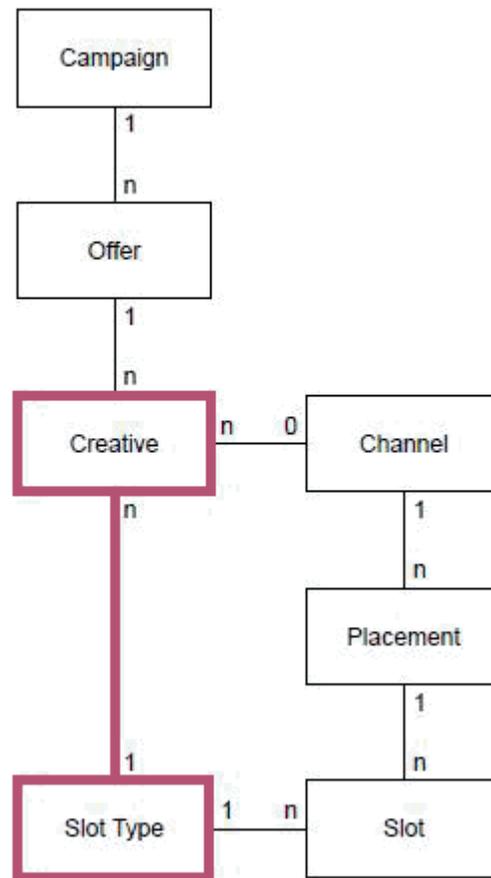
Channels perspective



Slots by Slot Type perspective



Creatives by Slot Type perspective



Perspectives are defined in the file `clm\Build\metadata\ref\config\perspectives.xml` (you can modify this if you use a metadata.module different from ref).

`perspectives.xml` is made of a `<perspectives>` root node which contains multiple perspectives.

Note: In this section, property entries shown in bold are required entries.

Properties of perspective:

- **name:** the name of the perspective
- one **root** node
- multiple level nodes (ordered)

Properties of root:

- **choiceGroupId:** the choice group that will be the root node of the tree
- **simpleViewCriteria** (see later in this section)
- **viewCriteria** (see later in this section)

Properties of level:

- **choiceGroupId**: the choice group that will be the node of the tree at that level
- **relationshipTypeId**: the relationship to follow between the previous level and this level
- **reversed**: whether to follow the relationship from `fromChoiceGroupId` to `toChoiceGroupId` (`reversed = false`) or from `toChoiceGroupId` to `fromChoiceGroupId` (`reversed = true`, the default).

For the cardinalities currently supported, *:1 and *:0, use `reversed=true` to display many child nodes under each node

- `simpleViewCriteria` (see later in this section)
- `viewCriteria` (see later in this section)

`simpleViewCriteria` and `viewCriteria` allow the application of a filter on which choices are displayed. You can specify multiple choices and the resulting filter will be the OR combination of all of them (that is, a choice will appear if it matches any of the criteria).

Properties of `simpleViewCriteria`:

- **attribute**: the choice attribute to filter on (of the choice group defined by `choiceGroupId` for this root or level)
- **operator**: the operator to use for filtering (see `searchCriteriaOperator` later in this section for a list of operators)
- **value**: the value of the choice attribute to filter on

Note: To specify criteria for date attributes, the value must be in the format `yyyy-mm-dd`, such as in:

```
<simpleViewCriteria attribute="startDate"  
operator="AFTER" value="2012-11-25"
```

Properties of `viewCriteria`:

- **name**: the id of a criteria defined on the choice group of the same `choiceGroupId` as this root or level

2.2.2.3 Choice Group and Relationship-Types XML Files

You can put as many xml configuration files in the perspectives configuration folder as you wish.

These configuration files let you specify choice groups and relationship types between choice groups.

Relationship types are owned by one side. In other words, to add, edit or delete a relationship between two choices, the user must be able to lock the choice owning that relationship.

Note: In the following lists, properties shown in bold mean that they are required.

Properties of choice groups:

- **id**: the id of the choice group. It must match the id in the Inline Service
- **name**: the name of the choice group. It will appear as such in the Decision Manager user interface

- **description:** the description of the choice group.
- **searchSortOrder:** the order in which this choice group will appear in the search dropdowns (sorted ascending based on this number)

If you do not put this property, then this choice group will not appear in the dropdowns

- **createSortOrder:** the order in which this choice group will appear in the create dropdown (sorted ascending based on this number)

If you do not put this property, then this choice group will not appear in the dropdown

- **attributes:** an array of attribute (see later in this section)
- **criteria:** zero to many criteria used for perspective filtering (see later in this section)

Properties of attributes:

- **id:** the id of the choice attribute. It must match the id in the Inline Service.
- **name:** the name of the choice attribute. It will appear as such in the Decision Manager user interface.
- **description:** the description of the choice attribute. This description will appear in tool tips in the user interface, and is a useful way to give advice to business users on how to enter values for this attribute.
- **searchCriteriaOperator:** the operator used for the advanced search.

If you do not put this property, then this attribute will not show up in the advanced search (unless you use the Add Field dropdown).

Valid values for strings:

- "<>" (Not equal to)
- "NOTBETWEEN" (Not between)
- "CONTAINS" (Contains)
- "ISNOTBLANK" (Is not blank)
- "=" (Equal to)
- "<" (Less than)
- ">" (Greater than),
- "ISBLANK" (Is blank)
- "<=" (Less than or equal to)
- ">=" (Greater than or equal to)
- "DOESNOTCONTAIN" (Does not contain)
- "STARTSWITH" (Starts with),
- "ENDSWITH" (End with)
- "BETWEEN" (Between)

Valid values for dates:

- "<>" (Not equal to)
- "ISBLANK" (Is blank)

- "ONORAFTER" (On or after)
- "BEFORE" (Before)
- "NOTBETWEEN" (Not between)
- "ISNOTBLANK" (Is not blank)
- "ONORBEFORE" (On or before)
- "AFTER" (After)
- "BETWEEN" (Between)
- "=" (Equal to)

Valid values for numbers:

- "<>" (Not equal to)
- "ISBLANK" (Is blank)
- ">" (Greater than)
- "<=" (Less than or equal to)
- ">=" (Greater than or equal to)
- "NOTBETWEEN" (Not between)
- "ISNOTBLANK" (Is not blank)
- "BETWEEN" (Between)
- "<" (Less than)
- "=" (Equal to)

The text within each set of parentheses shows what appears in the user interface.

- `genSysNote`: controls whether changing the value of such an attribute should trigger a "Set Attribute" audit trail entry, true or false, true by default
- `type` (see later in this section)
- `restriction` (see later in this section)

Properties of `type`:

- **name**: the type of the attribute, either "string", "date" or "number"
- **length**: for string types, the maximum number of characters that can be entered into the text control. This includes the characters representing the new line. If set to 0 or less, the `maxLength` is ignored. Note that in some browsers like Internet Explorer, new line is treated as two characters.
- **precision**: for number types, the precision is the total number of digits. It must be between 1 to 38.
- **scale**: for number types, the scale is the number of digits to the right of the decimal point. It can range from -84 to 127.
- **displayWidth**: the size of the text control specified by the number of characters shown. The number of columns is estimated based on the default font size of the browser.
- **required**: whether a non-null, non-empty value must be entered.

Properties of `restriction`:

- **kind**: the kind of type restriction, either "lov" (only for string attributes), "regexp" (only for string attributes), or "range" (only for number attributes).
- **minDefined**: for range type restrictions, whether the range has a min bound ("true" or "false")
- **maxDefined**: for range type restrictions, whether the range has a max bound ("true" or "false")
- **minInclusive**: for range type restrictions, whether the range min bound is included ("true" or "false")
- **maxInclusive**: for range type restrictions, whether the range max bound is included ("true" or "false")

(all other type restriction information is retrieved at runtime from the Inline Service)

Properties of relationship types:

- **id**: the id of the relationship type. You use this id in your Inline Service for methods such as `CLMChoiceBag.getRelatedChoiceIds()`.
- **fromChoiceGroupId**: the choice group that owns this relationship type.
- **toChoiceGroupId**: the choice group destination.
- **fromName**: the owner to destination relationship type name. You use this name in your Inline Service for methods such as `CLMChoiceBag.getRelatedChoiceIds()`.
- **toName**: the destination to owner relationship type name. You use this name in your Inline Service for methods such as `CLMChoiceBag.getRelatedChoiceIds()`.
- **fromDescription**: the owner to destination relationship type description.
- **toDescription**: the destination to owner relationship description.
- **cardinality**: From the owner to the destination, can be:
 - ***:1**
Each source has one destination
Each destination has 0 to n sources
 - ***:0**
Each source has 0 or 1 destination
Each destination has 0 to n sources
- **onDelete**: 'C' to cascade delete destination choices, 'O' to make destination choices orphans (that is, only delete the relationship). Explicitly:
 - **onDelete="C"**: Deleting a choice on the "zero or one" side of the relationship will attempt to also delete the choices on the "many" side
 - **onDelete="O"**: Deleting a choice on the "zero or one" side of the relationship will not delete the choices on the "many" side
- **propagateRules**: 'N' to not propagate rules (the default), 'D' to propagate rules from owner to destination, 'S' to propagate rules from destination to owner, 'B' to propagate rules in both directions. For more details, see the section "Propagation of Rules" in *Oracle Real-Time Decisions Base Application Installation and Reference Guide*.
- **propagateEvents**: 'N' to not propagate rules (the default), 'D' to propagate rules from owner to destination, 'S' to propagate rules from destination to owner, 'B' to propagate rules in both directions. For more details, see the section "Propagation

of Events" in *Oracle Real-Time Decisions Base Application Installation and Reference Guide*

Properties of criteria:

- **id**: the id of the criteria
- **xml**: the <ViewCriteria> xml of a view criteria defined on this choice group's VO (view object).

Projects are a special kind of choice group. You can specify additional attributes to projects by specifying a choice-group with id Project. Project attributes cannot be required and do not appear in the user interface. They only appear in the model layer, where they can be accessed by the Oracle RTD public Java APIs.

2.2.2.4 Security XML Files

By default, all security metadata is in a file called `security.xml` but you may create as many files containing security information as you want.

There are different root nodes that apply to security:

- <application-roles> contains the definition of application roles
- <enterprise-roles> contains the definition of enterprise roles, which are uploaded as WebLogic groups in a development environment, but are not meant to be used in a production environment
- <users> contains the definition of users, which are uploaded as WebLogic users in a development environment, but are not meant to be used in a production environment

<application-roles>

<application-roles> contains multiple <application-role>.

Each application role has a <name> and <permissions>.

<permissions> has multiple <permission>, each having a <resource-name>, a <resource-type> and <actions>.

The resource type can be perspective, choice group and project.

For the perspective resource type, the resource name is the name of the perspective as defined in `perspectives.xml`, or `_all_` to mean all perspectives.

For the perspective resource type, the only action is view, which means that the user having this application role will be able to see and select this perspective in the Decision Manager user interface.

For the choice group resource type, the resource name is the id of the choice group as defined in one of the other configuration files, or `_all_` to apply to all choice groups.

For the choice group resource type, the actions are a comma separated list of one or multiples of:

- **create**: to be able to create a choice of that choice group in a project
- **read**: to be able to view choices of that choice group
- **update**: to be able to edit a choice of that choice group in a project, and to be able to discard any changes (including addition and deletion) on a choice of that choice group
- **delete**: to be able to delete a choice of that choice group in a project

- own: to be able to edit (in the same project) and become owner of a choice that is owned by someone else

For the project resource type, the resource name must be `_all_`.

For the project resource type, the actions are a comma separated list of one or multiples of:

- create: to be able to create a project
- commit: to be able to commit a project
- discard: to be able to discard a project and all changes within it
- update: to be able to update a project
- read: to be able to log in to the application and to use the Decision Manager user interface

<enterprise-roles>

<enterprise-roles> has multiple <enterprise-role>.

Each <enterprise-role> has a name and <application-roles>, which is a comma separated list of application roles that are granted to members of that enterprise role.

<users>

<users> has multiple <user>.

Each <user> has a name, a display name, a description and credentials which is the password in an encrypted form. See previous sections for how to get the correct credential string.

Each <user> has <application-roles>, which is a comma separated list of application roles that are granted to members of that enterprise role.

2.2.2.5 Folder for Choice Group Images

For each choice group you have defined, add a 16x16 png image in the `images\group` folder (under `<metadata_modules_home>\<your_application>`) named after the choice group id.

The file must be called `<choiceGroupId>.png`.

Both parts of the file name are case-sensitive:

- `<choiceGroupId>` must exactly match the name of the choice group id, as set up in the choice group configuration xml file (see [Section 2.2.2.3, "Choice Group and Relationship-Types XML Files"](#))
- png must be all lower-case

2.2.2.6 Inline Service Folder

The RTD_Base_Marketing Inline Service released with the RTD for Marketing Optimization application is in the service folder under `<metadata_modules_home>\ref`, that is, in `clm\Build\metadata\ref\service`. This Inline Service is described in more detail in *Oracle Real-Time Decisions Base Application Installation and Reference Guide*.

2.2.3 Ant Tasks

After you have modified the metadata, you need to run the application generation tool `ant` to generate the application.

The `clm\Build` directory contains the ant tasks to perform the generation.

The ant targets are as follows:

- `ear` - creates EAR file for deployment in production mode
- `clean` - cleans projects
- `generate` - generates the application.
- `overwrite` - overwrites files with custom files.
- `clean-generated` - removes generated code

The ant option "generate" takes one parameter: `metadata.module`. The default value is `ref`. This specifies which subfolder of `clm\Build\metadata` should be used to generate the application.

For instance, to generate an application in `clm\Build\metadata\myapp`, call:

```
ant -Dmetadata.module=myapp generate
```

Then

```
ant ear (for deploying to production, otherwise deploy from JDeveloper)
```

The ant option "overwrite" copies the files in the `metadata.module overwrite` subfolder and copies them over any existing file. Use this if you are making changes to files that are generated by the generate task but want to keep the changes that were made. To do so, copy the file in the `overwrite` subfolder after you have applied your changes, following the same directory structure from the root (the root being where the file `clm.jws` is). You should then call `ant overwrite` after calling `ant generate`, in order to retain your changes.

If you have made changes to metadata that affect one of the files you have already overwritten, you need to manually apply these changes by merging these changes in the files in the `overwrite` folder.

`ant generate` will warn you if it is about to overwrite files that have been modified and should therefore be kept in the `overwrite` folder. To overwrite this warning, add `-Dchanges=overwrite` to the ant generate command line.

Notes:

- After generation, the database schema may have changed. You should then run "drop `ils.sql`" then "drop `core.sql`" (or drop and recreate the database user), then "load `core.sql`" then "load `ils.sql`". This will delete any choice you have in the database. To preserve choices, you can compare the previous and new sql and just make these changes. For instance, if you added a choice attribute, you can alter the table to add a column, and recreate the view associated with it.
- The files modified by application generation are usually in sub-directories called `ils`. The following files are modified by application generation and are NOT in `ils` sub-directories:
 - `clm\src\META-INF\jazn-data.xml` (you must include your users and roles in security xml configuration files)
 - `clm\ViewController\adfmsrc\config.xml` (you must edit `clm\Build\metadata\ref\config\config.xml` instead, replacing `ref` with your metadata module directory if different)

- `clm\ViewController\adfmsrc\perspectives.xml` (you must edit `clm\Build\metadata\ref\config\perspectives.xml` instead, replacing `ref` with your metadata module directory if different)
- `clm\ViewController\adfmsrc\oracle\rtd\clm\ui\DataBindings.x`

2.2.4 Java API

The behavior of the Oracle RTD Decision Management business layer can be further extended by the use of the Java API that comes with it. The javadoc for this Java API can be found in `clm\lib\clm-model-api-javadoc.jar`.

2.2.5 Application Extensions

This section contains the following topics:

- [Section 2.2.5.1, "Adding an Attribute to a Choice Group"](#)
- [Section 2.2.5.2, "Adding a Choice Group"](#)
- [Section 2.2.5.3, "Creating a Relationship Type Between Two Choice Groups"](#)
- [Section 2.2.5.4, "Modifying Perspectives"](#)

2.2.5.1 Adding an Attribute to a Choice Group

In order to add an attribute to a choice group, follow these steps:

- Modify the xml where this choice group is defined in folder `clm\Build\metadata\ref`.
- Run application generation again: in folder `clm\Build`, run `"ant generate"`.
- Since adding an attribute to a choice group modifies the database, you have to recreate the database. In order to do this, call `"drop ils.sql"` then `"drop core.sql"` then `"load core.sql"` then `"load ils.sql"` (these files are in `clm\Database\sql` and `clm\Database\sql\ils`). If you do not want to lose your database data, you can compare the sql to the previous one and alter the table to add the column and recreate the associated view.
- You can now deploy the new version of the Decision Management application using JDeveloper.
- You have to add the choice attribute to the choice group in your Inline Service as well. Make sure the id is the same as the one you entered in the xml metadata in the first step. Redeploy the Inline Service.

2.2.5.2 Adding a Choice Group

In order to add a choice group, follow these steps:

- Add an xml file for this new choice group in folder `clm\Build\metadata\ref`.
- Add an image for this choice group in folder `clm\Build\metadata\ref\images\group`.
- Run application generation again: in folder `clm\Build`, run `"ant generate"`.
- Since adding a choice group modifies the database, you have to recreate the database. In order to do this, call `"drop ils.sql"` then `"drop core.sql"` then `"load core.sql"` then `"load ils.sql"` (these files are in `clm\Database\sql` and `clm\Database\sql\ils`). If you do not want to lose your database data, you can

compare the sql to the previous one and add the new table, the new view and one row in the CHOICE_GROUP table.

- You can now deploy the new version of the Decision Management application using JDeveloper.
- You have to add the choice group in your Inline Service as well. Make sure the ids are the same as the ones you entered in the xml metadata in the first step. Make sure the choice group you add is under **CLM Base** choice group. Add the choice group in the **CLM ILS Choice Groups** application parameter. Redeploy the Inline Service.

2.2.5.3 Creating a Relationship Type Between Two Choice Groups

In order to create a relationship type between two choice groups, follow these steps:

- Edit `relationship-types.xml` in folder `clm\Build\metadata\ref` and add the new relationship type.
- Run application generation again: in folder `clm\Build`, run "ant generate".
- Since adding a choice group modifies the database, you have to recreate the database. In order to do this, call "drop ils.sql" then "drop core.sql" then "load core.sql" then "load ils.sql" (these files are in `clm\Database\sql` and `clm\Database\sql\ils`). If you do not want to lose your database data, you can compare the sql to the previous one and add one row in the RELATIONSHIP_TYPE table.
- You can now deploy the new version of the Decision Management application using JDeveloper.
- You can (but do not have to) add choice attributes in both choice groups in the Inline Service to follow the relationship. For an example, see how Offer has a "campaign" choice attribute and a "creatives" choice attribute. Redeploy the Inline Service.

2.2.5.4 Modifying Perspectives

In order to modify perspectives, follow these steps:

- Edit `perspectives.xml` in folder `clm\Build\metadata\ref\config`.
- Run application generation again: in folder `clm\Build`, run "ant generate".
- You can now deploy the new version of the Oracle RTD Decision Management application using JDeveloper.

2.3 Miscellaneous

This section contains the following topics:

- [Section 2.3.1, "Configuring Logs"](#)

2.3.1 Configuring Logs

Oracle RTD Decision Management uses Java Logging API. There are two ways to configure Java Logging API

(<http://java.sun.com/j2se/1.5.0/docs/guide/logging/overview.html>):

- Update global logging configuration of the JRE which is used to start WebLogic:
`JAVA_HOME/jre/lib/logging.properties`

- Create separate logging properties file (for example `c:\src\clm\logging.properties`) and pass it in the `-Djava.util.logging.config.file` argument to the Weblogic.Server startup command.

To pass this argument append the line:

```
set JAVA_OPTIONS=%JAVA_OPTIONS%
-Djava.util.logging.config.file=
c:\src\clm\logging.properties
```

to

```
C:\Oracle\Middleware\user_projects\domains\<RTDCLM_
domain>\bin\setDomainEnv.cmd
```

Sample logging.properties file:

```
# Specify the handlers to create in the root logger
.level= INFO
handlers = weblogic.logging.ServerLoggingHandler
# Register handlers for the oracle.rtd.clm. and its child loggers
oracle.rtd.clm.handlers = java.util.logging.FileHandler,
java.util.logging.ConsoleHandler, weblogic.logging.ServerLoggingHandler
oracle.rtd.clm.useParentHandlers = false
oracle.rtd.clm.level = ALL
#Console handler
java.util.logging.ConsoleHandler.level = INFO
java.util.logging.ConsoleHandler.formatter = java.util.logging.SimpleFormatter
# Set the default logging level for new FileHandler instances
weblogic.logging.ServerLoggingHandler.level = ALL
#File handler
java.util.logging.FileHandler.pattern = %h/java%u.log
java.util.logging.FileHandler.limit = 50000
java.util.logging.FileHandler.count = 1
java.util.logging.FileHandler.formatter = java.util.logging.SimpleFormatter
java.util.logging.FileHandler.level=ALL
```

This configuration logs message with level INFO to the Weblogic Server Startup Console and all messages to the `java*.log` file in the user home directory. To change log level for Oracle RTD Decision Management applications replace `oracle.rtd.clm.level = ALL`

with `oracle.rtd.clm.level = SEVERE | WARNING | INFO | CONFIG | FINE | FINER | FINEST.`

Additional info can be found at the following site:

http://download.oracle.com/docs/cd/E14571_01/web.1111/e13739/logging_services.htm

