IBM Operational Decision Manager
Version 8 Release 6

Configuring Operational Decision Manager on WebLogic

Note

Before using this information and the product it supports, read the information in "Notices" on page 61.

This edition applies to version 8, release 6, modification 0 of Operational Decision Manager and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Chapter 1. Configuring Operational Decision Manager on Oracle WebLogic Server

Configuring Rule Execution Server on WebLogic Server

To use Rule Execution Server on a new instance of Oracle WebLogic Server, you must establish your database credentials, deploy the provided archives for this server, and perform a number of configuration tasks.

Operational Decision Manager supports Oracle WebLogic Server 11g, 10.3.6, and 12c.

Before you start: Opening the administration console

To configure Rule Execution Server on Oracle WebLogic Server, you must follow specific steps.

About this task

To configure Rule Execution Server on the WebLogic Server application server, you work in the WebLogic Server Administration Console.

These instructions are designed for users who are familiar with their database and the application server. However, some notes or examples are included throughout to help nonexpert users.

Procedure

1. After WebLogic Server is installed, create a domain.

   Refer to the WebLogic Server documentation if necessary.
2. Use the Windows **Start** menu to start the WebLogic Server administration server from the new domain.
3. Use the Windows **Start** menu again to start the WebLogic Server Administration Console.
4. Log in to the Administration Console by using the user name and password that you defined when you created the WebLogic Server domain.
5. Start your database.

   Note: The beginner examples use the Derby database and not the embedded database. Beginners should therefore download and start the Derby database.

What to do next

The following table summarizes the steps to configure Rule Execution Server on WebLogic Server, depending on the type of persistence.

Installation Steps	Persistence		
	File	**Data source**	**JDBC**
"Step 1: Selecting and applying the persistence type" on page 2	✔	Default persistence mode	✔

Installation Steps		Persistence		
		File	Data source	JDBC
"Step 2: Restricting database user permissions" on page 3		✓	✓	✓
"Step 3: Establishing the database credentials" on page 4		Not applicable	✓	✓
"Step 4: Creating a data source and connection pool" on page 4		Not applicable	✓	✓
"Step 5: Configuring security" on page 5	"Adding groups" on page 5	✓	✓	✓
	"Adding users" on page 6	✓	✓	✓
"Step 6: Deploying the XU RAR" on page 10		✓	✓	✓
"Step 7: Deploying the Rule Execution Server management EAR file" on page 11		✓	✓	✓
"Step 8: Creating a Rule Execution Server database schema" on page 12	"Creating a database schema from the Rule Execution Server console" on page 13	Not applicable	✓	✓
	"Creating a database schema by using the SQL scripts" on page 16	Not applicable	✓	✓
"Enabling Anonymous Admin Server Lookup for MBeans" on page 8		✓	✓	✓
"Step 9: Deploying the hosted transparent decision service EAR" on page 17		Optional	Optional	Optional
"Step 10: Verifying the configuration" on page 18		Optional	Optional	Optional

Related concepts:
Troubleshooting Rule Execution Server on Oracle WebLogic Server

Step 1: Selecting and applying the persistence type

You can change the default `datasource` RuleApp and Java™ XOM persistence settings by running an Ant script that generates a new Rule Execution Server management archive.

Typically, you do this if you are in development mode. This step does not apply to beginners who work with the embedded Derby database.

For you to change the persistence settings, the distribution provides an Ant script in the <*ODM_InstallDir*>/executionserver/bin/ressetup.xml file. Use it to create a new instance of the Rule Execution Server management archive and, in the case of a Java EE application server, the execution unit (XU).

Solaris users

If you use file-based persistence on Solaris, your file system must support all characters used in directory and file names that are present in the ruleset path (RuleApp name and ruleset name). Set the **LANG** system property with the encoding that is compatible with your package and rule names, for example en_US.UTF-8.

Decision Warehouse

If you select the file persistence type for RuleApps, you cannot use Decision Warehouse.

MySQL persistence

If you choose to use MySQL as a persistence back end, add or set the following properties in the MySQL configuration file: my.ini on Windowsor my.cnf on UNIX operating systems:

```
sql-mode=STRICT_ALL_TABLES
max_allowed_packet=1073741824
```

For more information about these settings, see the MySQL 5.0 reference manual: 5.1.7. Server SQL Modes and 5.1.4 Server System Variables.

Step 2: Restricting database user permissions

If Rule Execution Server data is stored in a database, the database administrator might require that you provide the specific permissions to access the database.

Note: This step applies when database access needs to be restricted. If you manage the database yourself (for example, you use an embedded database for test purposes) or if you do not need further restrictions, skip this step and proceed to the next configuration step.

Connection to the Rule Execution Server database, as established in the data source credentials, and any subsequent requests to the database are handled through a database user. This database user (name and password), for example resdbUser, is defined by the database administrator and has no relation to the standard Rule Execution Server groups.

The following table gives the typical list of permissions that the database administrator must define on the Rule Execution Server database, with attention given to the type of operations. Some supported databases do not require all these permissions.

	Operation	
Database permission	**Browse and edit rulesets and RuleApps**	**Create the Rule Execution Server schema**
CREATE ANY INDEX	Not required	Required
DROP ANY INDEX	Not required	Required
CREATE ANY SEQUENCE	Not required	Required

Database permission	Operation	
	Browse and edit rulesets and RuleApps	Create the Rule Execution Server schema
DROP ANY SEQUENCE	Not required	Required
SELECT ANY SEQUENCE	Required	Not required
CREATE ANY TABLE	Not required	Required
DROP ANY TABLE	Not required	Required
INSERT ANY TABLE	Required	Not required
SELECT ANY TABLE	Required	Not required
UPDATE ANY TABLE	Required	Not required
DELETE ANY TABLE	Required	Not required
CREATE ANY TRIGGER	Not required	Required
CREATE ANY VIEW	Not required	Required
DROP ANY VIEW	Not required	Required

Step 3: Establishing the database credentials

If you set the persistence mode to a data source or to a JDBC connection, you must establish the credentials of the database that is dedicated to Rule Execution Server.

These credentials are required to establish the data source. See "Step 4: Creating a data source and connection pool." If no database exists, create one now. For some types of databases, you can also create the database when you establish the connection to the data source.

Note: If you set the persistence mode to `file`, you can skip all the database-related tasks and proceed directly to "Step 5: Configuring security" on page 5.

After the database is created, go to "Step 8: Creating a Rule Execution Server database schema" on page 12 to learn how to create the schema to include tables and views. You can create the database schema either from the Installation Settings wizard or by running the provided SQL scripts. You can find these scripts in the `<InstallDir>/executionserver/databases` directory.

Step 4: Creating a data source and connection pool

Create a JDBC data source and connection pool.

Before you begin

You create the data source and connection pool in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 1 for details.

Procedure

1. In the WebLogic Server administration console, in **Domain Configurations** under **Services** , click **Data Sources** (or **JDBC** > **Data Sources**).
2. In the Summary of JDBC Data Sources page, on the **Configuration** tab, in the Data Sources table, click **New** > **Generic Data Source** (or just **New**).

3. In the Create a New JDBC Data Source page, name your data source (for example, `Rule Execution Server Data Source`) and type `jdbc/resdatasource` for the JNDI name.
4. Select the type from the **Database Type** drop-down menu and click **Next**. For example, select **Derby**.
5. In the next panel, select a driver from the **Database Driver** drop-down menu to connect to the database and click **Next**.

   The best practice is to use the XA database driver. The **Transaction Options** box displays information.
6. If you created a connection pool with a non-XA driver and you want to allow your data source to be involved in XA transactions, select **Emulate Two-Phase Commit**, and then and click **Next**. The **Supports Global Transactions** option is selected by default.
7. Define the **Connection Properties** and click **Next**.

   For example, for a Derby database:

Table 1. Connection properties for a Derby database

Property	Value
Database name	`resdb;create=true`
Host name	`localhost`
Port number	`1527`
Database user name	`resdbUser`
Password	Type `resdbUser` and confirm it.

8. In the Test Database Connection page, click **Test Configuration** to ensure that the connection pool is operational, and then click **Next**. If the console displays the following message, your connection pool is ready to be used.

   `Connection test succeeded.`
9. In the next panel of the Create a New JDBC Data Source page, in the Servers table, select the **AdminServer** check box and click **Finish**.

Results

The display returns to the Summary of JDBC Data Sources page, which shows your settings and this message: `All changes have been activated. No restarts are necessary.`

Step 5: Configuring security

You control access to Rule Execution Server and enforce security by defining groups, users, and associated roles.

Adding groups

Create the groups that have access to Rule Execution Server.

Before you begin

You add groups in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 1 for details.

About this task

You control access to Rule Execution Server and enforce security by defining user groups.

The following table summarizes the main groups and their associated default user and password.

Group	Use	Default user and password
resAdministrators	Gives a user full administrator rights for the following actions: • Access and use the Rule Execution Server console to populate the database schema • Deploy, browse, and modify RuleApps • Monitor the decision history, purge, and back up the history • Run diagnostics and view server information	`resAdmin` `resAdmin`
resDeployers	Gives a user the following rights: • Deploy, browse, and modify RuleApps • Test rulesets	`resDeployer` `resDeployer`
resMonitors	Gives a user the following rights: • View RuleApps • Monitor decision history and access Decision Center reports	`resMonitor` `resMonitor`

Procedure

1. At the bottom of the Home Page of the WebLogic Server administration console, click **Security Realms**.
2. In the Summary of Security Realms page, in the **Realms** table, click **myrealm**.
3. In the Settings for myrealm page, click the **Users and Groups** tab and then click the **Groups** tab.
4. In the Groups page, click **New**.
5. In the Create a New Group page, enter `resAdministrators` for the group name and `Rule Execution Server administrator` for the description. Click **OK**.

   The console displays the following message: `Group created successfully`
6. Create the `resMonitors` and the `resDeployers` groups in the same way.

Adding users

After the groups are created, create the users and add them to the groups.

Before you begin

You add users in the WebLogic Server Administration Console. See “Before you start: Opening the administration console” on page 1 for details.

Procedure

1. At the bottom of the Home Page of the WebLogic Server administration console, click **Security Realms**.

If you are just finished creating groups on the Settings for myrealm page, you do not have to go back to the Home Page: just click the **Users** tab (step 3).

2. In the Summary of Security Realms page, under **Realms** click **myrealm**.
3. In the Settings for myrealm page, click the **Users and Groups** tab and then click **Users**.
4. In the Users table, click **New**.
5. In the Create a New User page, enter `resAdmin` for the name and a password with a minimum of 8 characters and one digit (such as `resAdmin1`).

   Important:

   The user names must be an exact match of the strings `resAdmin`, `resDeployer`, and `resMonitor` because the `weblogic.xml` file defines these strings in the security role assignments. You can find the `weblogic.xml` file in the `jrules-res-management-WL10.ear/jrules-res-management.war/WEB-INF/` or `jrules-res-management-WL12.ear/jrules-res-management.war/WEB-INF/` directory.

```
<weblogic-web-app .......>
        <security-role-assignment>
                <role-name>resAdministrators</role-name>
                <principal-name>resAdmin</principal-name>
        </security-role-assignment>
        <security-role-assignment>
                <role-name>resDeployers</role-name>
                <principal-name>resDeployer</principal-name>
                <principal-name>resAdmin</principal-name>
        </security-role-assignment>
        <security-role-assignment>
                <role-name>resMonitors</role-name>
                <principal-name>resMonitor</principal-name>
                <principal-name>resDeployer</principal-name>
                <principal-name>resAdmin</principal-name>
        </security-role-assignment>

...
</weblogic-web-app>
```

   If you want to change the user name, you can change the `principal-name` in the `weblogic.xml` file.

   If you want different names but do not want to change the deployment descriptor, you can select **Custom Roles** as the security model when you deploy the Rule Execution Server management console application.
6. Confirm the password and click **OK**.

   The `User created successfully` message is displayed and the `resAdmin` user is listed in the Users table.
7. In the **Name** column, click `resAdmin` to open the Settings for resAdmin page.
8. Click the **Groups** tab.

   Now, to add members to the `resAdministrators` group, you must add the group to the user definition.
9. Under Parent Groups, scroll down to select `resAdministrators` in the Available column and click the arrow to move `resAdministrators` to the Chosen column.
10. Click **Save**.

These instructions do not apply if you are deploying Rule Execution Server on a WebLogic Server cluster and you have enabled the administration port on your management server. In this case, it is not possible to remove the link between the `resAdmin` user and the `Administrators` group.

Note:

You can reassign the user name for the Rule Execution Server console in WebLogic Server. To do so, edit the `weblogic.xml` file in the `jrules-res-management-WL10.war` or `jrules-res-management-WL12.war` file and create a new user with the same name in the WebLogic Server administration console.

11. Repeat steps 4 through 10 on page 7 to create the users and add them to the groups as shown in the following table.

User	Password	Groups
resMonitor	resMonitor1	resMonitors
resDeployer	resDeployer1	resDeployers, resMonitors
resAdmin	resAdmin1	resAdministrators, resDeployers, resMonitors, Administrators

What to do next

If you want to give the `resDeployer` and `resMonitor` users the rights to run Rule Execution Server diagnostics, see "Allowing non-administrators to run diagnostics" on page 9 for instructions.

Enabling Anonymous Admin Server Lookup for MBeans

To give the MBean API read-only access to WebLogic Server MBeans, you must enable the `anonymous admin lookup` option.

Before you begin

You enable anonymous lookup in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 1 for details.

About this task

To use Rule Execution Server with WebLogic Server, you must activate the **Anonymous Admin Lookup Enabled** option in WebLogic Server. This option provides the MBean API with read-only access to WebLogic Server MBeans. Registration of full Rule Execution Server management and XU MBeans requires this access.

If the **Anonymous Admin Lookup Enabled** option is disabled when you start the deployment of the management EAR file, the following exception is raised:

```
javax.naming.NoPermissionException: User <anonymous> does not have permission on weblogic.management
```

An exception is also raised when you start the deployment of the XU resource archive (RAR file).

These instructions explain how to verify or set the **Anonymous Admin Lookup Enabled** option through the WebLogic Server administration console. You can also use to command line to set the **SecurityConfigurationMBean.AnonymousAdminLookupEnabled** attribute:

```
-Dweblogic.management.anonymousAdminLookupEnabled=true
```

Procedure

To check or modify the `anonymous admin lookup` option:

1. On the Home Page, click **Domain**.
2. In the Settings for <your_domain_name> page, click the **Security** tab.
3. On the **General** page, select **Anonymous Admin Lookup Enabled** and click **Save**.
4. Restart WebLogic Server.

Allowing non-administrators to run diagnostics

After you have created users and added them to groups, you might want to allow non-administrators to run diagnostics in Rule Execution Server.

Before you begin

You work in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 1 for details.

About this task

By default, the `resDeployer` and `resMonitor` users do not have WebLogic Server administrator rights, which they need to run the diagnostics. The following procedure shows how to add a Java Management Extensions (JMX) policy for these users to access all the MBeans. For the **JMX Policy Editor** to be available on the Policies page, you must first activate a specific option to protect JMX access.

Procedure

1. From the breadcrumbs at the top of the console, or from the Home Page, open the Summary of Security Realms page, and in the Realms table, click **myrealm**.
2. On the Settings for myrealm page, on the **Configurations** > **General** tab, select the **Use Authorization Providers to Protect JMX Access** check box.
3. Click **Save** and restart the server.

   Tip: It is not necessary to restart the Administration Console.
4. After the server is restarted, click the **Roles and Policies** tab, and then the **Realm Policies** tab.
5. Under **Policies**, click **JMX Policy Editor**.
6. On the JMX Policy Editor page, check that **GLOBAL SCOPE** is selected and click **Next**.
7. In the next panel, check that **ALL MBEAN TYPES** is selected and click **Next**.
8. In the Attributes and Operations table, select **Operations: Permission to Invoke** and click **Create Policy**.
9. Click **Add Conditions**, select **Role** from the **Predicate List**, and click **Next**.
10. Type `Admin` in the **Role Argument Name** field and click **Add**, and then click **Finish**.
11. On the Edit JMX Policies page, click **Add Conditions**, select **Group** from the **Predicate List** list, and click **Next**.

12. Add the groups that do not have administrator privilege and whom you allow to run the diagnostic tests. For example, to add all the `resDeployers` and `resMonitors` groups follow these steps:
    a. Type `resDeployers` in the **Group Argument Name** field and click **Add**.
    b. Type `resMonitors` in the **Group Argument Name** field and click **Add**.
    c. Click **Finish**

    The policy conditions and the overridden policy are shown:
    ```
    Group: resDeployers or resMonitors

    Role: Admin
    ```
13. Click **Save**.
14. Restart WebLogic Server.

Step 6: Deploying the XU RAR

The execution unit (XU) is a resource adapter for Java EE-based application servers. It is supplied as a RAR archive that you must deploy on your application server.

Before you begin

To deploy the resource adapter for the execution unit, you work in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 1 for details.

Before you deploy the XU RAR file, make sure that the **Anonymous Admin Lookup Enabled** option is turned on, otherwise you get an error message. See "Enabling Anonymous Admin Server Lookup for MBeans" on page 8.

About this task

You can use the upload directory of WebLogic Server to deploy the modules at start up. The upload directory depends on the server configuration; by default it is named `/autodeploy`. The directory is in the list of directories that are shown in the WebLogic Server administration console.

Note: In some cases, because of your application constraints, you might have to deploy the XU inside the application. Choose the appropriate deployment mode of the XU: either embed it into the EAR file or deploy it as a global connector. Be aware of the following consequences:

- When the XU is deployed as a global connector:
  - The deployed Java™ EE applications might use its third-party libraries (such as ASM) instead of the libraries that are deployed in the application server.
  - Use the `parent last` setting for the XU Java EE application if your Java EE application does not support the version of the third-party libraries that is distributed with Decision Server. If you cannot use the `parent last` setting, you might have to embed the XU into the EAR that executes the rules.
- If you choose an embedded XU packaging, use the `parent last` setting for the code library if the version of the third-party libraries that is deployed at the level of the application-server code library is not compatible with the XU.

Tip: When the default configuration in shared mode is not appropriate for your use case, you can configure Rule Execution Server so that it is scoped to a single Java™ EE application. See Configuring Rule Execution Server in scoped mode.

You can find the name of your server by opening **Environment** in the **Domain Structure** and clicking **Servers**.

Procedure

1. At the bottom of the Home Page, click **Deployments**.
2. In the **Summary of Deployments** page, under **Deployments**, click **Install**.
3. In the **Install Application Assistant**, click the link **upload your file(s)**.
4. Click **Browse** next to **Deployment Archive** and upload one of the RAR files, and click **Next**.
   - `<InstallDir>/executionserver/applicationservers/WebLogic10/jrules-res-xu-WL10.rar`
   - `<InstallDir>/executionserver/applicationservers/WebLogic12/jrules-res-xu-WL12.rar`
5. Make sure that the **jrules-res-xu-WL10.rar** or **jrules-res-xu-WL12.rar** option is selected and click **Next**.

   The process might take a few moments.
6. Select **Install this deployment as an application** and click **Next**.
7. Make sure that **Use the defaults defined by the deployment's targets** is selected and click **Finish**.

   The resource adapter is shown as **Active** in the Deployments table.

Step 7: Deploying the Rule Execution Server management EAR file

After you have defined security settings and deployed the XU RAR file, you can deploy the Rule Execution Server management EAR file on Oracle WebLogic Server.

Before you begin

To deploy the management archive, you work in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 1 for details.

Before you deploy the management EAR file, make sure that the **Anonymous Admin Lookup Enabled** option is turned on, otherwise you get an error message. See "Enabling Anonymous Admin Server Lookup for MBeans" on page 8.

Procedure

1. On the Home Page of the WebLogic Server administration console, click **Deployments**.
2. In the Summary of Deployments page, click **Install**.

   The Install Application Assistant opens in a separate tab.
3. In Install Application Assistant, click **upload your file(s)** in the Note.
4. Click **Browse** next to **Deployment Archive**, upload one of the EAR files, and click **Next**.
   - `<ODM_InstallDir>/executionserver/applicationservers/WebLogic10/jrules-res-management-WL10.ear`
   - `<ODM_InstallDir>/executionserver/applicationservers/WebLogic12/jrules-res-management-WL12.ear`

   In the next panel, the EAR file that you uploaded is selected.

5. Click **Next**. The upload might take a few moments.
6. Select **Install this deployment as an application** and click **Next**.
7. In the next panel, set the values as follows:
   - If you want to use the default users that are specified in the deployment descriptor of the Rule Execution Server management console application, select **DD Only: Use only roles and policies that are defined in the deployment descriptors**.
   - Otherwise, select **Custom Roles: Use roles that are defined in the Administration Console; use policies that are defined in the deployment descriptor** for defining other users in the administrative console.
8. If you selected **Custom Roles: Use roles that are defined in the Administration Console; use policies that are defined in the deployment descriptor.** in step 7, you must proceed as follows:
   a. Click **Next**.
   b. Keep the **Yes, take me to the deployment's configuration screen** option enabled and click **Finish**.

      After a few minutes, the Settings for <management_EAR_file_name> page is displayed.
   c. Click the **Security** tab.
   d. Under Enterprise Application Scoped Roles, click **New**.
   e. In the Create Application Scoped Roles page, enter `resAdministrators` as the name of the new role, and then click **OK**.
   f. Repeat this step e to create two other roles: `resDeployers` and `resMonitors`.
   g. Click **resAdministrators** in the **Enterprise Application Scoped Roles** table.
   h. Click **Add Conditions**, select **Group** in the **Predicate List** and click **Next**.
   i. Enter `resAdministrators` for the **Group Argument Name**.
   j. Click **Add**, then click **Finish** and then click **Save**.
   k. Go back to the **Roles** page.

      Use the breadcrumbs at the top of the page for easy navigation.
   l. Repeat steps d to f to add the `resDeployers` and then the `resMonitors` groups as the role condition
9. Accept the other default settings and click **Finish**.

Results

In the Summary of Deployments page, the state of the EAR file is now **Active**.

Related concepts:

"WebLogic Server clusters" on page 21
You can deploy Rule Execution Server on a Oracle WebLogic Server cluster. A single management module is necessary for all the execution units (XU).

Step 8: Creating a Rule Execution Server database schema

Provided that you have established database credentials, you can create the schema for the Rule Execution Server database from the command editor or from the Rule Execution Server console.

Creating a database schema from the Rule Execution Server console

To create the Rule Execution Server database schema, you can run the database scripts from the Rule Execution Server console and use the Installation Settings wizard if you work on Windows and in other supported distributed environments.

Installation Settings wizard overview:

On Windows and in distributed environments only, you can use the Installation Settings wizard of the Rule Execution Server console to create a database schema with the necessary database tables and views.

To open the Installation Settings wizard, you must sign in to the Rule Execution Server console with administrator rights. The default user with these rights is `resAdmin`. For more information about setting passwords and the permission to access the Installation Settings wizard, see "Step 5: Configuring security" on page 5.

To configure the database to store managed Java XOM, you follow the procedure twice, with a different target database each time.

The Installation Settings wizard creates all the necessary tables for Rule Execution Server and for Decision Validation Services, even if you do not have a license for Decision Validation Services.

The combination of persistence settings for RuleApps and managed Java XOMs affects how you use the Installation Settings wizard.

- If the RuleApp persistence and the Java XOM persistence are both set to `file`: No wizard is presented.
- If the RuleApp persistence and the Java XOM persistence are both set to `datasource` or JDBC: When you sign in to the Rule Execution Server console, the **RuleApp persistence details** part of the Installation Settings wizard opens for you to create the schema for RuleApps and the Decision Warehouse trace. After you have completed this step, the **Java XOMs persistence details** part of the wizard is displayed for you to configure the database for Java XOM persistence.
- If the RuleApp persistence is set to `file` and the Java XOM is set to `datasource` or JDBC: You see only **Java XOMs persistence details** and you cannot use the Decision Warehouse.
- If the RuleApp persistence is set to `datasource` and the Java XOM is set to `file` or is not defined: When you sign in to the Rule Execution Server console, only **RuleApp persistence details** openss for you to create the schema for RuleApps. The **Java XOMs persistence details** part of the wizard is not shown.

The following table summarizes the cases.

Persistence		RuleApps	
		file	datasource/JDBC
Java XOMs	file	No Installation Settings wizard	**RuleApps persistence details** only
	datasource	**Java XOMs persistence details** only No Decision Warehouse	Complete Installation Settings wizard

Opening the Rule Execution Server console:

To open the Rule Execution Server console and the Installation Settings wizard, you must sign in with `resAdministrators` rights.

About this task

To access the Installation Settings wizard, you must first open and sign in to the Rule Execution Server console.

Procedure

1. Open the Rule Execution Server console in a web browser by typing `res` at the root URL on the host computer.

   `http://<hostname>:<port>/res`.

   - If your browser is not running on the same host as the application server, replace `localhost` with the address of the computer.
   - If the web application is mapped to a host that is defined on a different port than the default of the server, change the port number to the host port number, usually `7001`.

   For example: `http://localhost:7001/res`
2. Sign in to the Rule Execution Server console as the administrator.

   The default credentials are `resadmin` for the user ID and `resAdmin1` for the password.

Results

If you are using database persistence and the database schema is empty, the Installation Settings wizard opens and you can use it to complete the installation.

Step 1: Welcome to the Installation Settings wizard:

If you open the Rule Execution Server console with a data source or a JDBC connection as the persistence setting and an empty database, the Installation Settings wizard opens.

The wizard can display two parts:
- It starts with **RuleApp persistence details** if you set `datasource` or `jdbc` persistence for RuleApps, whatever the persistence type for managed Java XOMs.

- **Java XOMs persistence details** is shown after **RuleApp persistence details** if you set `datasource` persistence for both RuleApps and Java XOMs. In this case, you go through the same steps twice.

  The wizard starts with **Java XOMs persistence details** if you set the persistence type to `file` for RuleApps and to `datasource` for managed Java XOMs.

Both parts of the wizard are similar and you use them in the same way:

1. The Welcome page provides the following information:
   - Persistence details about the type of the database, such as information about the driver and the JDBC URL.
   - A brief description of the purpose of the Installation Settings wizard.
   - A diagnostic report that provides information about why the persistence check failed, because you have not created the database tables yet.

   Click **Next**.
2. "Step 2: Choose the database schema."
3. "Step 3: Review the database schema"
4. "Step 4: The Installation Settings wizard report" on page 16

Step 2: Choose the database schema:

Choose the database schema. Specific settings are available for DB2®. Optionally, you can select a customized SQL script.

Procedure

1. In the **Database schema selected** field, select an available database schema type.

   A type that corresponds to the type of database that you are using is selected by default, but you can choose another type from the drop-down list.

   If you select a `db2` schema, an extra field becomes available for you to enter the name of the buffer pool, which is used to create the Decision Warehouse table space. This buffer pool must have a page size of 32 Kb. Check the DB2 documentation for information about how to create a 32 Kb buffer pool.

   Note: The scripts for creating the Decision Warehouse database on DB2 are written for databases that use automatic storage. When you use the Installation Settings wizard, you create the database for both the Rule Execution Server and Decision Warehouse. Therefore, your database must use automatic storage.

   If you have not configured your DB2 database to use automatic storage, you cannot use the Rule Execution Server console to create the schema.

   Decision Warehouse table spaces are always created, even if you have not installed Decision Validation Services.
2. Optional: Select **custom** if you want to use a customized SQL script, then click **Browse** to select the location of the custom script.
3. Click **Next** to review the database schema.

Step 3: Review the database schema:

After you have selected a database schema, you have access to a choice of options for the schema.

Before you begin

First, ensure that you have a backup of database resources.

About this task

After you have selected a database schema, you confirm the creation of a schema for Rule Execution Server. You can also use SQL drop statements that flush data from an existing table and view the SQL statements.

Procedure

1. Select from the following options:
   - **Create SQL schema "resdbUser"**: Select this option to run the SQL statement for the schema type that you selected in "Step 2: Choose the database schema" on page 15.
   - **Keep drop SQL statements**: Select this option to flush data from an existing Rule Execution Server database.
   - **Show SQL statements**: Select this option to display the SQL statements.
2. Click **Execute** to start the selected options.

Step 4: The Installation Settings wizard report:

Finally, after you have selected and confirmed the schema, the Installation Settings wizard reports the status.

Procedure

1. Click **Show execution details** to view the list of executed SQL statements.
2. Click **Finish**.

   If you have just worked in **RuleApps persistence details** and the persistence setting for managed Java XOMs is `datasource`, the **Java XOMs persistence details** part of the wizard displays now for you to repeat the procedure.

Creating a database schema by using the SQL scripts

Use SQL scripts to create a dedicated schema if you are using a database as your persistence layer.

The script that creates the database schema is named `repository_<DatabaseName>.sql`. You can find SQL scripts in this directory: `<InstallDir>/executionserver/databases`.

Note:

If you want to use Decision Warehouse, you can also create the database table by running the `trace_<DatabaseName>.sql` script. If Java XOMs are also stored in a database, you must create these tables by running the `xomrepository_<DatabaseName>.sql` script.

If you run the scripts in a command-line processor, you must log in with the credentials that you use for the data source for Rule Execution Server.

To import and run the SQL scripts, use any tool that can handle SQL. The following database tools are provided.

Database	Database tool
IBM® DB2	DB2 command line processor
Derby	ij command line processor
MySQL	mysql command line processor
Oracle	sqlplus command line processor
Postgre QL	Postgre SQL command-line tool
SQL Server	Query Tool
Sybase	isql command line processor

To access the database, the database user must secure the following credentials and rights:

- A user ID
- A password
- Complete privileges on the tables, and a view of the schema (create, insert, delete)
- Privileges for index creation (`create index`)
- On Oracle, the additional `create trigger` and `create sequence` privileges.

If you use an Oracle database, run all the scripts in the SQL Plus client.

Install a database client for the database that you use. For more information, refer to the documentation of your database system.

Step 9: Deploying the hosted transparent decision service EAR

You can optionally deploy the hosted transparent decision service on your application server.

Before you begin

To deploy the archive for hosted transparent decision services, you work in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 1 for details.

About this task

You must deploy the hosted transparent decision service on the same node as the execution unit (XU).

Procedure

1. At the bottom of the Home Page, click **Deployments**.
2. In the **Summary of Deployments** page, under **Deployments**, click **Install**.
3. In the **Install Application Assistant**, click the **upload your file(s)** link.
4. Click **Browse** next to **Deployment Archive**, upload one of the EAR files, and click **Next**.
   - `<InstallDir>/executionserver/applicationservers/WebLogic10/jrules-res-htds-WL10.ear`
   - `<InstallDir>/executionserver/applicationservers/WebLogic12/jrules-res-htds-WL12.ear`

5. Make sure that **jrules-res-htds-WL10.ear** or **jrules-res-htds-WL12.ear** is selected and click **Next**.

   The process might take a few moments.
6. Select **Install this deployment as an application** and click **Next**.
7. Accept the default settings and click **Finish**.

   At the end of the deployment, the Summary of Deployments page is displayed and the EAR file is shown as **Active** in the Deployments table. Otherwise, select the check box next to the artifact and select **Start** > **Servicing all requests**.
8. Set the **`ruleset.xmlDocumentDriverPool.maxSize`** ruleset property to the appropriate value. See Setting the ruleset.xmlDocumentDriverPool.maxSize property.

What to do next

For more information about checking that the hosted transparent decision services are deployed successfully, see Viewing and managing transparent decision services in the Rule Execution Server console online help.

Step 10: Verifying the configuration

You can verify that Rule Execution Server has been successfully configured by running the Rule Execution Server diagnostics.

About this task

If the diagnostics run before any execution units (XU) have been started, the test is passed and a message states that no XUs are initialized.

Important: For a scalable number of users to access resources through the Java components, the Java EE Connector Architecture (JCA) assigns the task of implementing connection pools to application server vendors.

Procedure

1. Open the Rule Execution Server console by typing `res` at the root URL on the host computer: For example: `http://localhost:7001/res`

   If your browser is not running on the same host as the application server, replace `localhost` with the address of the computer.

   If the web application is mapped to a host that is defined on a different port than `7001`, change the port number accordingly.
2. Sign in to the Rule Execution Server console.
3. Click the **Diagnostics** tab.
4. Click **Run Diagnostics**.

   A report shows all the diagnostic tests that have just been run, with a check mark next to each test. Click **Expand All** to show details about all the tests.

Optional configuration steps

You can enhance your configuration with additional options, such as multiserver configuration, or deploying onto a server cluster.

Repackaging the Rule Execution Server archive by using Ant

To repackage a Rule Execution Server archive to configure the Rule Execution Server instance, you can use an Ant task, provided that you set up the Ant task environment.

Before you begin

Before you run the **res-setup** Ant task, you must set up the Ant task environment correctly. For more information, see Setting up your environment to use Ant.

About this task

By default, persistence is set to `datasource`. To change the persistence type, you can use an Ant task. An Ant script is provided with the distribution for this purpose. The script creates new archives that use a specific persistence mode.

The following procedure uses Ant to repackage the archives to change the persistence mode to `file`.

Procedure

Write the code that creates a new XU `.RAR` file and management `.EAR` file to set `file` persistence:

```
ant -Dxu.in=../applicationservers/WebLogic10/jrules-res-xu-WL10.rar
-Dxu.out=myxu.rar
-Dconsole.ear.in=../applicationservers/WebLogic10/jrules-res-management-WL10.ear
-Dconsole.ear.out=mymanagement.ear
-Dpersistence.type=file -f ressetup.xml setup
```

Configuring Rule Execution Server for different environments

General configuration guidelines apply if you want to enable Rule Execution Server on different environments in a single cell.

Before you begin

These instructions assume that you do not configure multiple instances of Rule Execution Server in a single cell.

About this task

Most likely, the development of your business rule management system (BRMS) requires more than a single deployment of Rule Execution Server. The development lifecycle of a business rule application is similar to any other software development process: it includes stages for implementation, testing, deployment, and maintenance. At the very least, you are likely to need an environment for your development team, one for your QA team, and another one for in-production applications. When you configure Rule Execution Server in a single cell, it is good practice to isolate the rulesets that you use on each server and ensure that the execution units (XUs) do not interfere with each other.

Consider the following guidelines to set up your different environments in a single cell.

Procedure

1. Set up different data sources.
2. Deploy and configure a XU for each environment.

3. Deploy the Rule Execution Server console for each environment.
4. To set up a data source for each environment, use unique JNDI names.

   For example: `jdbc/resdatasourceEnv1` and `jdbc/resdatasourceEnv2`
5. To deploy the Rule Execution Server console for each environment, proceed as follows:
   a. Modify the deployment descriptor of the Rule Execution Server EAR or WAR management archive: in the `web.xml`file, uncomment the **`JMX_XU_QUERY_PART`** parameter and specify `xuName=xuEnv1`.
   b. Deploy the Rule Execution Server EAR or WAR file to the server.

      In the resource, reference settings in the application server:
      1) Set the JNDI for the data source as:`jdbc/resdatasourceEnv1`.
      2) Set the JNDI name for the XU a: `eis/ConnectionFactoryEnv1`.
   c. Repeat a and b to deploy the Rule Execution Server console for the other environments.
6. Restart the node agents after you complete the configuration.
7. Call the XU instances to register the XU with the Rule Execution Server console.

Rule Execution Server deployment on cluster environments

You can use Rule Execution Server in a cluster or other multiserver environments.

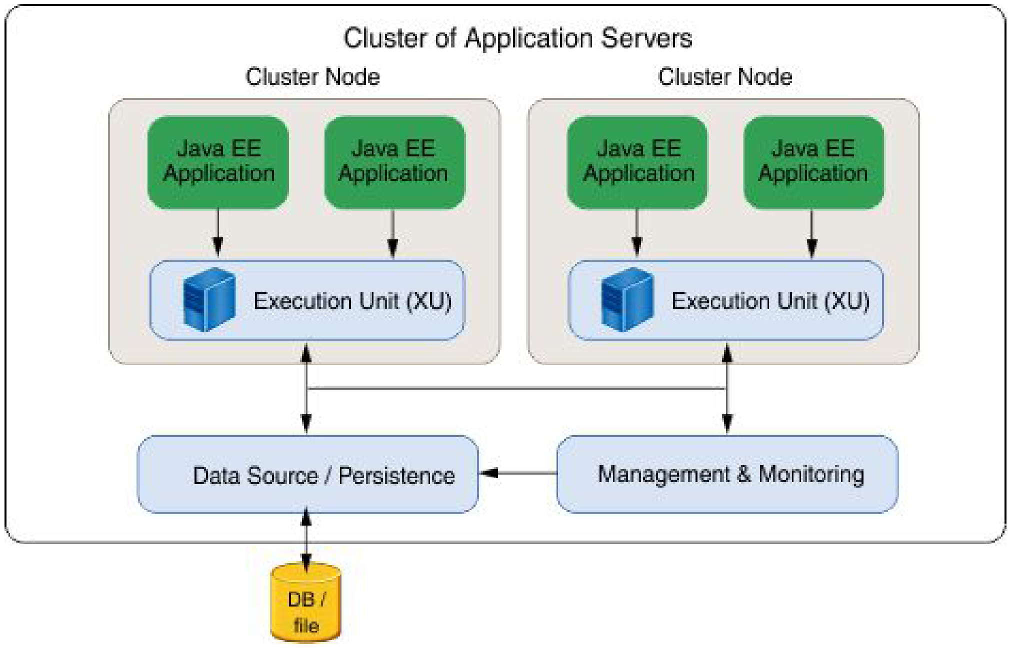
Cluster configuration and topology:

When you deploy Rule Execution Server onto server clusters, each node hosts one execution unit (XU), which is used only by a local rule session. The cluster topology significantly affects the notification mechanism.

Within the Java EE framework, clusters provide mission-critical services to ensure minimal downtime and maximum scalability. A cluster is a group of application servers that transparently run your Java EE application as if it were a single entity.

Cluster implementations on Java EE application servers come with their own set of terminology. Each of the following terms are important to understand how your cluster performs:
- Cluster and component failover services
- HTTP session failover
- Single points of failure in a cluster topology

On a cluster configuration, deploy an execution unit (XU) on each node. There is one XU for each node of a cluster. Use the administration console of application servers to handle cluster deployment. A XU instance can be used only by a local (same node) rule session. The rule session and the XU communicate through direct Java method calls, so the XU does not require serialization.



A cluster that uses Rule Execution Server involves a collaboration between the Rule Execution Server MBeans. The topology of the cluster has significant influence on the management of the notification mechanism when a resource is changed.

The management model is likely to use several times a basic scenario of a distributed notification mechanism within a cluster to interact with the various execution unit (XU) instances. A XU message-driven rule bean (MBean) is deployed with the XU to collaborate with the Rule Execution Server JMX infrastructure.

The following sequence applies:

1. A management client sets a resource on a ruleset MBean.
2. The ruleset makes a query to the MBean server to retrieve all the XU instances in the cluster. This operation requires a specific implementation for each application server.
3. A notification is sent to each instance.

WebLogic Server clusters:

You can deploy Rule Execution Server on a Oracle WebLogic Server cluster. A single management module is necessary for all the execution units (XU).

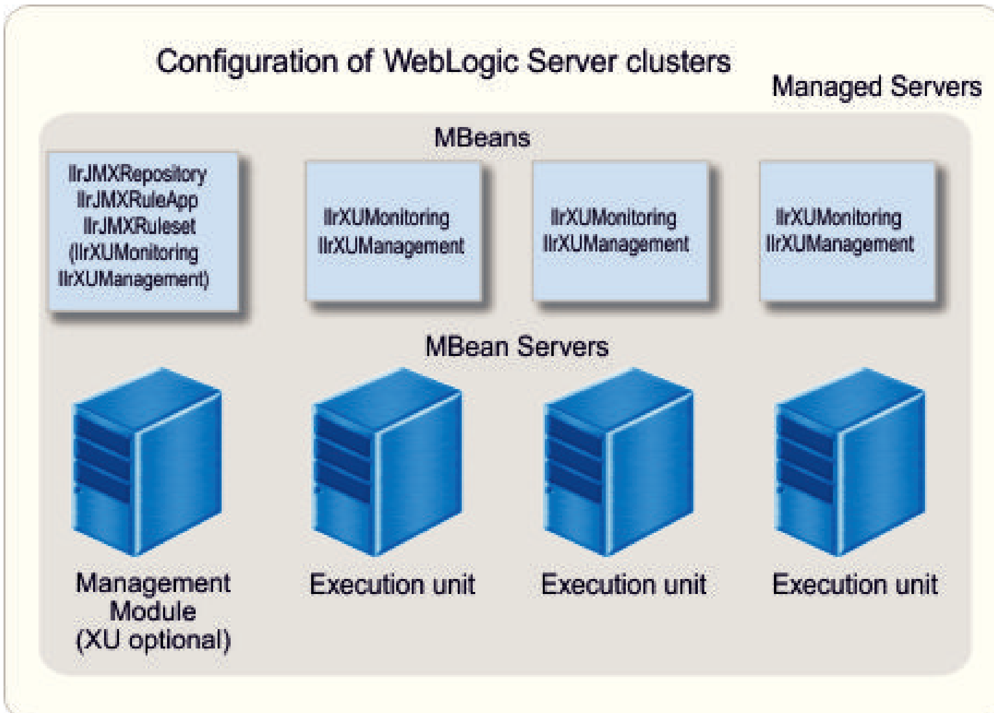
When you deploy the Rule Execution Server components on WebLogic Server, you must deploy the management module on a single server in the WebLogic Server domain. You can deploy the execution unit (XU) and the management module on the same server, but only a single management module is necessary to manage the deployment of a XU on each server in the cluster.

Follow these guidelines:

1. Define the cluster as the target for the data source and the XU.
2. Define one of the servers in the cluster as the target for the management module when you deploy the `jrules-res-management-WL10.ear` or `jrules-res-management-WL12.ear` file.

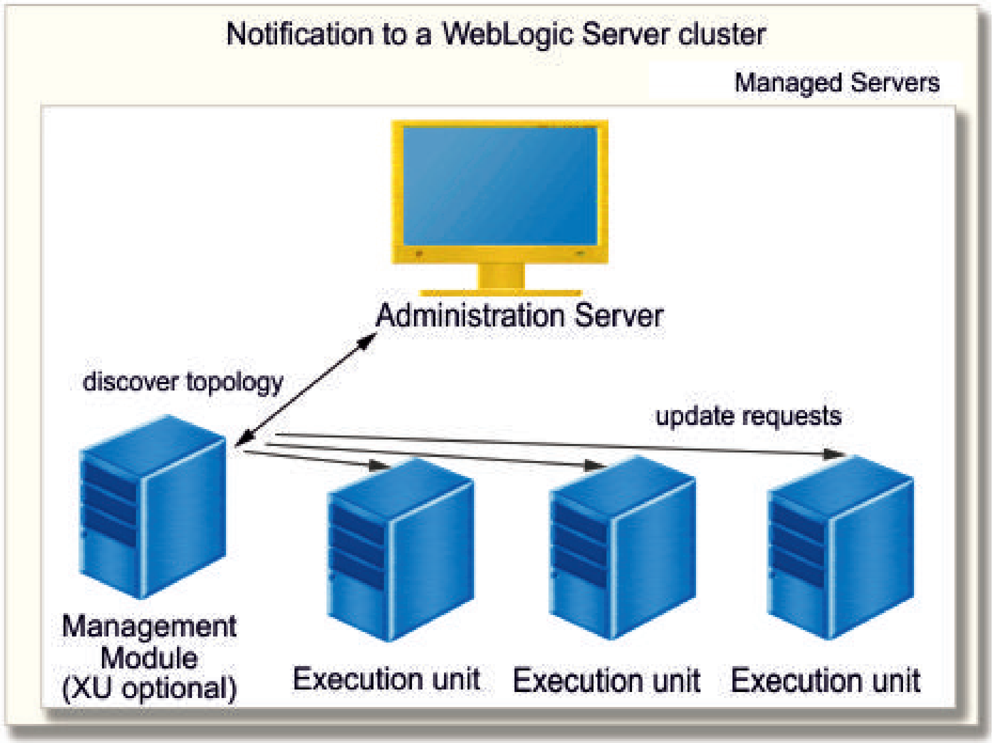
3. Enable anonymous lookup in the security options in the WebLogic Server administration console.
4. Restart all the servers.

The default location for the logs of the managed servers is: `<WL_HOME>/common/nodemanager/NodeManagerLogs`. You can use these logs to resolve any JNDI naming issue of the data source.



The notification mechanism works by transmitting messages to all the instances of a XU on the managed servers. The management model queries the administration server to determine the list of managed servers, it then queries the local MBean server on each managed server to notify all the XU instances.

You do not have to install the WebLogic Server administration console on the administration server. You install it only once in the domain to keep the integrity of the displayed information. Notification of updates is propagated only to the XUs and not to other consoles. Therefore, multiple consoles deployed in that domain are not synchronized. For further information about WebLogic Server clusters, see Understanding WebLogic Server Clustering.



For more information, refer to:

- WebLogic JNDI
- Cluster-Wide JNDI Naming Service

Related tasks:

"Step 7: Deploying the Rule Execution Server management EAR file" on page 11
After you have defined security settings and deployed the XU RAR file, you can deploy the Rule Execution Server management EAR file on Oracle WebLogic Server.

Configuring the Decision Center consoles on Oracle WebLogic Server

To use the Decision Center Enterprise or Business consoles on a new instance of WebLogic Server, you must deploy the provided archives for this application server and perform a number of configuration tasks.

Operational Decision Manager supports Oracle WebLogic Server 11g, 10.3.6, and 12c.

Before you start: Opening the administration console

To configure Decision Center on Oracle WebLogic Server, you must open the Administration Console and follow specific steps. Some of these steps are mandatory, while others apply in certain cases only.

About this task

To configure Decision Center on the WebLogic Server application server, you work in the WebLogic Server Administration Console.

These instructions are designed for users who are familiar with their database and the application server. However, some notes or examples are included throughout to help nonexpert users.

You can complete the configuration of your application server either with the Installation Settings wizard of the Decision Center console or with Ant scripts.

After the installation is finished, you can start Decision Center. For more information, see Verifying your configuration of Decision Center. The first time that you open Decision Center, it does not contain rule projects. You must publish a project from Rule Designer. For more information, see Publishing a project to Decision Center.

Tip:

If you want to work with rule projects that were created with JRules 7 before Operational Decision Manager 8.6, you must migrate the database schema as described in Migrating the business user database.

Procedure

1. After WebLogic Server is installed, create a domain.

   Refer to the WebLogic Server documentation if necessary.
2. Use the Windows **Start** menu to start the WebLogic Server administration server from the new domain.
3. Use the Windows **Start** menu again to start the WebLogic Server Administration Console.
4. Log in to the console by using the user name and password that you defined when you created the WebLogic Server domain.
5. Start your database.

   Note: The beginner examples use the Derby database and not the embedded database. Beginners should therefore download and start the Derby database.

What to do next

The following table summarizes the steps to configure Decision Center on WebLogic Server.

Step		Required
"Step 1: Restricting database user permissions" on page 25		
"Step 2: Creating a data source and connection pool" on page 26		
"Step 3: Configuring security" on page 27	"Groups and permissions" on page 27	
	"Creating groups in WebLogic Server" on page 28	
	"Creating users in WebLogic Server" on page 30	

Step		Required
"Step 4: Deploying the Decision Center EAR file" on page 31		✔
"Step 5: Verifying the deployment of the Decision Center consoles" on page 32		Optional
"Step 6: Completing Decision Center configuration" on page 33	"Completing the configuration from the Decision Center Enterprise console" on page 33	✔ Use one of the two methods to complete the installation. Required for distributed platforms only.
	"Completing the configuration by using Ant tasks" on page 38	
Verifying your configuration		✔

Step 1: Restricting database user permissions

Decision Center data is stored in a database. The database administrator might require that you provide the specific permissions that you need when accessing this database.

Note: This step applies when database access needs to be restricted. If you manage the database yourself (for example, you use an embedded database for test purposes) or if you do not need further restrictions, skip this step and proceed to the next configuration step.

Connection to the Decision Center database, established in the data source credentials, and any subsequent requests to the database are handled through a database user. This database user (name and password), for example `rtsdbUser`, is defined by the database administrator and has no relation to the standard Decision Center groups.

The following table gives the permissions that the database administrator must define on the Decision Center database, with attention given to the type of operations that you want to perform. Some supported databases do not require all the above permissions.

	Operation			
Database permission	**Use Decision Center**	**Create the database schema by using the Decision Center console or Ant tasks**	**Modify the database schema by using the Decision Center console or Ant tasks**	**Migrate the database schema**
ALTER TABLE	Not required	Not required	Required	Required
CREATE INDEX	Not required	Required	Required	Required
CREATE ROLE	Not required	Not required	Not required	Required
CREATE SEQUENCE	Not required	Required	Required	Required
CREATE TABLE	Not required	Required	Required	Required

	Operation			
Database permission	**Use Decision Center**	**Create the database schema by using the Decision Center console or Ant tasks**	**Modify the database schema by using the Decision Center console or Ant tasks**	**Migrate the database schema**
CREATE VIEW	Not required	Required	Required	Required
DROP INDEX	Not required	Not required	Required	Required
DROP SEQUENCE	Not required	Not required	Required	Required
DROP TABLE	Not required	Not required	Required	Required
DROP VIEW	Not required	Not required	Required	Required
INSERT TABLE	Required	Required	Required	Required
SELECT SEQUENCE	Required	Required	Required	Required
SELECT TABLE	Required	Required	Required	Required
UPDATE TABLE	Required	Required	Required	Required

Step 2: Creating a data source and connection pool

You must create a data source and connection pool in WebLogic Server to enable Decision Center.

Before you begin

You create the data source and connection pool in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 23 for details.

About this task

Before you create a data source, your database must be running. For a list of supported databases, consult the support details on the IBM support site.

Procedure

To create a data source and connection pool:

1. On the Home Page, in the Domain Configurations section, under **Services**, click **Data Sources** (or **JDBC** > **Data Sources**).

   The Summary of JDBC Data Sources page opens.
2. In the Summary of JDBC Data Sources page, under **Data Sources**, click **New** > **Generic Data Source** or just **New**.
3. Give a name to your data source and type `jdbc/ilogDataSource` for the JNDI name.

   For example, you can name the data source `Decision Center Data Source`.
4. Select a database type and click **Next**. For example, select **Derby**.
5. In the next panel, select a database driver and click **Next**.

   Tip: Best practice is to use a non-XA driver.

6. In the panel that displays transaction options, keep the default options and click **Next**.
7. Define the connection properties and click **Next**.

   For example, Derby users can enter the following values:

Table 2. Connection properties for a Derby database

Property	Value
Database name	`rtsdb;create=true`
Host name	`localhost`
Port number	`1527`
Database user name	`rtsUser`
Password	Type `rtsUser` and confirm it.

8. In the next page, click **Test Configuration** to ensure that the connection pool is operational.

   If the message `Connection test succeeded.` is displayed, your connection pool is ready to be used.
9. Click **Next**.
10. In the Create a New JDBC Data Source page, in the **Servers** table, select **AdminServer** and click **Finish**.

    The message `All changes have been activated. No restarts are necessary.` Is displayed and the Data Sources table lists the data sources that you created.

Step 3: Configuring security

You control access to Decision Center and enforce security by defining groups, users, and associated roles.

Groups and permissions

Enable the definition of the user name, passwords, and groups in text files without encryption.

Any user of Decision Center must belong to at least one of these mandatory groups:

- `rtsAdministrator`
- `rtsConfigManager`
- `rtsInstaller`
- `rtsUser`

Adherence to these groups determines what parts of Decision Center a user can access. You must create all these groups in WebLogic Server. For testing purposes, also create a default user and password for each of these groups.

In addition, if you want to perform the Decision Center permissions tutorial in your own installation, you must create two custom groups: `Validator` and `Eligibility`.

For more in-depth information on the Decision Center groups, see Groups.

This table summarizes the mandatory and custom groups and their associated role, default user, and password.

Group	Use	Default user/password
rtsAdministrator	Mandatory, gives the user administrator access.	rtsAdmin/rtsAdmin1
rtsConfigManager	Mandatory, gives the user configuration manager access.	rtsConfig/rtsConfig1
rtsUser	Mandatory, gives a user standard access.	rtsUser1/rtsUser1
rtsInstaller	Mandatory, gives the user access to the Installation Settings wizard.	rtsAdmin/rtsAdmin1
Validator	Optional custom group, used in the Decision Center permissions tutorial.	Val/Validator1
Eligibility	Optional custom group, used in the Decision Center permissions tutorial.	Eli/Eligibility1

Creating groups in WebLogic Server

Create groups and enable the definition of the user name, passwords, and groups in text files without encryption.

In WebLogic Server, you create user groups in three steps.

1. "Adding all groups to the security realm." This step is mandatory for all the groups that are used in Decision Center.
2. "Declaring global roles" on page 29. This step is mandatory for all the groups that are used in Decision Center.
3. Upload groups to the database. You do this later, in the Installation Settings wizard. For more information, see Completing the configuration from the Decision Center Enterprise console.

Adding all groups to the security realm:

Adding groups to the security realm is mandatory for all the groups that are used in Decision Center.

Before you begin

You create groups in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 23 for details.

Procedure

To add a group to the security realm:

1. In the WebLogic Server administration console, go back to the Home Page.
2. At the bottom of the Home Page, click **Security Realms**.
3. In the Summary of Security Realms page, in the **Realms** table, click the name **myrealm**.

   Do not just click the check box, click the name.
4. In the Settings for myrealm page, click **Users and Groups**.
5. Click the **Groups** tab and in the Groups page, click **New**.

6. In the Create a New Group page, add the `rtsAdministrator` group and type Decision Center `administrator` for the description, and click **OK**. Leave **Provider** set to `DefaultAuthenticator`.
7. When you have added the `dcAdministrator` group, repeat steps 5 on page 28 and 6 to add the following group names and their descriptions.

Group name	Description
rtsConfigManager	Decision Center configuration manager
rtsInstaller	Decision Center installer
rtsUser	Decision Center user
Validator	Optional Decision Center tutorial group
Eligibility	Optional Decision Center tutorial group

Your groups are added to the Groups table.

Declaring global roles:

After you have added mandatory groups to the security realm, you must declare global roles. Declaring global roles is mandatory for all the groups that you use in Decision Center.

Before you begin

You declare the global roles in the WebLogic Server Administration Console. See “Before you start: Opening the administration console” on page 23 for details.

About this task

Because you have just created groups, you should already have the Settings for myrealm page displayed. If not, click **Security Realms** at the bottom of the Home Page and then click **myrealm**.

Procedure

To declare global roles:

1. In the Settings for myrealm page, click **Roles and Policies**.
2. In the **Realm Roles** tab, in the Roles tables, expand **Global Roles** and click **Roles**.
3. On the Global Roles page, click **New**.
4. In the Create a New Role for this Realm page, under **Role Properties**, type the `rtsAdministrator` group name that you created and click **OK**.
5. In the Global Roles window, click the role name that you have just created.

   Do not select only the check box, click the name. You might have to click **Next** at the bottom of the table to see the new roles.
6. Click **Add Conditions**.
7. In **Predicate List**, select **Group** and click **Next**.
8. For **Group Argument Name**, specify the same name as the role, and click **Add**, **Finish**, and **Save**. For example, `rtsAdministrator`.
9. Use the breadcrumbs at the top of the page to go back to the Global Roles page.
10. Repeat steps 3 through 9 to define the remaining global roles and, optionally, the Validator and Eligibility roles for the tutorial:

Tip: Use the breadcrumbs at the top of the page for easy navigation.
- rtsConfigManager
- rtsInstaller
- rtsUser
- Validator
- Eligibility

What to do next

To use the Decision Center permissions mechanism, you must also upload groups to the database. See Completing the configuration from the Decision Center Enterprise console.

Creating users in WebLogic Server

Create users and add them to the groups that you have previously defined.

Before you begin

You create users in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 23 for details.

About this task

The following table gives a suggested configuration where you create a default user for each of the basic groups. Users `Val` and `Eli` are necessary only if you want to follow the Decision Center permissions tutorial in your installation.

Table 3. Suggested user names

User	Description	Password	Is added to groups...
rtsAdmin	Decision Center administrator	rtsAdmin1	rtsAdministrator, rtsInstaller
rtsConfig	Decision Center configuration manager	rtsConfig1	rtsConfigManager
rtsUser1	Decision Center user	rtsUser1	rtsUser
Eli	Decision Center tutorial user	Eligibility1	Eligibility, rtsUser
Val	Decision Center tutorial user	Validator1	Validator, Eligibility, rtsUser

Procedure

To add users:
1. If necessary, go back to the Home Page of the Administration Console.
2. At the bottom of the Home Page, click **Security Realms**.
3. In the Summary of Security Realms page, under **Realms**, click **myrealm**.

   Do not select only the check box, click the name.
4. In the Settings for myrealm page, click **User and Groups** and then click the **Users** tab.
5. In the Users page, click **New**.

6. In the Create a New User page, enter the user name, a description, the password, and confirm the password.

   For guidelines, see Table 3 on page 30.

   When you click **OK**, the following message is shown at the top of the page: `User created successfully`.
7. Repeat steps 5 on page 30 and 6 for the other users in the Table 3 on page 30 list.

Results

The Users table shows the new users.

Adding users to groups in WebLogic Server

After you create the users, you add them to the groups that you created earlier.

Before you begin

To add users to groups, you work in the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 23 for details.

About this task

If you are just finished creating users, the **Users** tab is still displayed. Otherwise, display it as follows:

1. At the Home Page, click **Security Realms**.
2. In the Summary of Security Realms page, under **Realms** click the name **myrealm**.
3. In the Settings for myrealm page, click **User and Groups**.

Procedure

To add users to groups:

1. In the Users page, click the user name that you want to add to the group.

   For example, click `rtsAdmin`. Do not only select the check box, click the name.
2. On the Settings for <user_name> page, click the **Groups** tab, then select `rtsAdministrator` in the **Available** column and click the right arrow to move `rtsAdministrator` to the **Chosen** column.
3. Click **Save**.
4. Use the breadcrumbs at the top of the console to go back to the Users table.
5. Repeat steps 1 through 4 for all the users in the Table 3 on page 30 list.

What to do next

You have configured the security settings, you can deploy the Decision Center EAR file.

Step 4: Deploying the Decision Center EAR file

After you have configured security settings, you can deploy the packaged Decision Center EAR file on WebLogic Server.

Before you begin

You deploy the EAR file from the WebLogic Server Administration Console. See "Before you start: Opening the administration console" on page 23 for details.

About this task

Deploying the Decision Center EAR file sets the persistence locale. After you save a rule to the database, you must no longer change the persistence locale. If you want to install Decision Center in a language other than English, take note of the instructions that are provided in Step 4: Set the persistence locale.

Procedure

To deploy the EAR file:

1. At the bottom of the Home Page, click **Deployments**.
2. In the Summary of Deployments page, under **Deployments**, click **Install**.
3. In the Install Application Assistant, click the **upload your file(s)** link.
4. Click **Browse** next to **Deployment Archive**, upload one of the following files, and click **Next**.
   - `<ODM_InstallDir>/teamserver/applicationservers/WebLogic10/jrules-teamserver-WL10.ear`
   - `<ODM_InstallDir>/teamserver/applicationservers/WebLogic12/jrules-teamserver-WL12.ear`
5. In the next panel, check that **jrules-teamserver-WL10.ear** or **jrules-teamserver-WL12.ear** is selected, and then click **Next**.
6. Select **Install this deployment as an application** and click **Next**.
7. In the Optional Settings page, make sure that the following options are selected.
   a. For Security, select **DD Only: Use only roles and policies that are defined in the deployment descriptors**.
   b. For Source accessibility, select **Use the defaults defined by the deployment's targets**.
8. Click **Finish**.

   After a few moments, the Summary of Deployments page is displayed again and the uploaded archive is set to **Active**.
9. If not, select the check box next to **jrules-teamserver-WL10.ear** or **jrules-teamserver-WL12.ear**, and then click **Start** > **Servicing All Requests**.

What to do next

Next, you verify the deployment of the Decision Center Enterprise console.

Step 5: Verifying the deployment of the Decision Center consoles

After you have finished configuring Decision Center for your application server, verify that you have deployed the archive successfully.

About this task

You start your application server and then use your web browser to open the Decision Center Enterprise console.

Procedure

1. Make sure that your application server is running.
2. Start a new browser instance and enter the default URL to access Decision Center in a web browser.
   - The default URL to access Decision Center is `http://localhost:<PORT_NUMBER>/teamserver`. Set *<PORT_NUMBER>* to the port number of your web application. If your browser is not running on the same host as the application server, replace `localhost` with the address of the host.
   - Starting from WebSphere® Application Server V8.0, each console has a different default URL.
     - For the Enterprise Console: `http://localhost:<PORT_NUMBER>/teamserver`
     - For the Business Console: `http://localhost:<PORT_NUMBER>/decisioncenter`
   - If your browser is not running on the same host as the application server, replace `localhost` with the address of the host.
   - If your web application is mapped to a host on a port that is different from the port number shown, change the port number to your host port number.

   The Decision Center log in page opens in your browser.
3. Sign in with `rtsAdministrator` rights to start testing.

   For example, `rtsAdmin` and `rtsAdmin`.

Step 6: Completing Decision Center configuration

After you have created a data source, defined security settings, and deployed the Decision Center EAR file, you complete the configuration either from the Decision Center console or by running Ant tasks.

Completing the configuration from the Decision Center Enterprise console

After you have deployed the Decision Center archive to your application server, you can work from the Decision Center Enterprise console to complete or modify the configuration.

Installation Settings wizard overview:

You use the Installation Settings wizard in the Decision Center console to create or modify the database schema, set up message files or groups, or change the persistence locale or configuration parameters.

The Installation Settings wizard opens automatically when you start the Decision Center console to complete an installation.

You can also open the Installation Settings wizard by clicking **Configure** > **Installation Settings Wizard** in the Decision Center console after you have completed your initial installation. If you open Decision Center after you have followed the steps to install the module, only the **Install** tab is available. For more information, see Opening the Decision Center consoles.

Note: To access the Installation Settings wizard, you must have both administrator privileges and the `rtsInstaller` role when you sign in.

You use the Installation Settings wizard for the following actions.

Table 4. Actions in the Installation Settings wizard

Action	Description
Configure the database	Mandatory when you complete the configuration with a database on a distributed platform. For more information, see Step 1: Configure the database.
Set up message files	Mandatory during the installation only if you have some custom rule model extension files. For more information, see Step 2: Set up message files.
Set up groups	You must set up the same groups that are declared in the application server if you want to use the Decision Center security and permissions mechanisms. For more information, see Step 3: Set up groups.
Change the persistence locale	Mandatory if the persistence locale is different from the locale `en_US`. For more information, see Step 4: Set the persistence locale.
Change configuration parameters	Optional. You change some configuration parameters when you customize Decision Center. For more information, see Step 5: Set configuration parameters.

After you have completed the installation, Decision Center is ready to use but does not contain rule projects. If you open Decision Center and no rule projects are available, a message in the **Configure** tab informs you that no project has been found and that you should either publish a rule project by using Rule Designer or contact the administrator.

If you see this message, you must publish a rule project from Rule Designer. For more information, see Publishing a project to Decision Center.

More information about using the Installation Settings wizard is available from the Decision Center console online help. To access the online help, click **Help** in the top banner after you have signed in to Decision Center.

Note: If you have rule projects that were created with a previous product version, refer to the Migrating topics on how to upgrade the Decision Center database schema.

Step 1: Configure the database:

You use the Installation Settings wizard to configure the database.

About this task

You store the extensions to the Decision Center rule model in two XML files:
- Model description: This file usually has the file name extension `.brmx`.
- Initialization of enumerations and hierarchies: This file usually has file name extension `.brdx`.

For more information about defining common model extensions, see the customization topics.

Procedure

1. When the Installation Settings wizard opens in Decision Center, click **Next**.
2. Select one of the extension files.
   - **Default extensions** (already selected)

- **Custom extensions (brmx/brdx)**, or
- **Custom extensions (Zip)**

3. Click **Generate SQL** to generate the script that creates the database tables, which are based on the contents of your rule model files.
4. After the script is generated, select the **Execute the SQL script** check box, and then click **Next**.

Step 2: Set up message files:

Message files contain the display text that is associated with the extensions to the rule model contained in the `.brmx` and `.brdx` files.

About this task

You can find the default message file in `<ODM_InstallDir>/teamserver/bin/defaultextensionmessages_<LOCALE>.properties`.

If you use the default rule model when you create your database, the default message file is automatically sent to the database. To upload your own message files, use the Installation Settings wizard as explained below.

You must have a message file for each locale that you use. Message files are identified by their locale. The contents of the message files must respect the ISO-LATIN-1 standard.

Procedure

To declare a message file in the Installation Settings wizard:

1. Click **New**.
2. Enter a locale.
3. Browse to the location of the message file for this locale.
4. Click **Apply**.

Results

If Decision Center supports this locale, the Installation Settings wizard assigns a locale code so that you can identify it.

Example

For example:

```
status=Status
effectiveDate=Effective Date
expirationDate=Expiration Date
new=New
defined=Defined
```

Step 3: Set up groups:

In addition to creating groups in your application server when you set up security access, you must use the Setup Groups page in the Installation Settings wizard to upload groups to the database.

Before you begin

Before you set up groups in the Enterprise console, make sure to add all the groups that you want to see in the available list when enforcing project security or setting permissions in Decision Center. For more information, refer to the topics on Groups and Permissions in the Decision Center help.

About this task

You set up groups only if you want to use the Decision Center project access and permission mechanisms.

Tip: In Decision Center, the groups are the roles in the application server, **not** the groups defined in the user registry. Decision Center uses the group information to verify whether a user belongs to a role in the application server.

You do not have to upload the `rtsAdministrator` or `rtsInstaller` group. The administrator group has access to everything, and an installer user must belong to another group.

You use the Setup Groups page in the Installation Settings wizard to upload the default groups for `rtsUser` and `rtsConfigManager`, and any custom groups, such as Validator and Eligibility if you want to follow the permissions tutorial.

Procedure

To set up groups:
1. Click **New**.
2. Type the group name.
3. Click **Apply**.
4. Repeat steps 1 to 3 for each group.
5. When you have added all the groups, proceed in one of the following ways:
   - Click **Next** if you want to set a different persistence locale, or configuration parameters.
   - Click **Finish** if you do not want to change these settings.

Step 4: Set the persistence locale:

The persistence locale determines the language in which you store rules in the Decision Center database.

About this task

You set the locale when you deploy the Decision Center EAR file to your application server. As a consequence, you store the rules in the database in the locale of the Decision Center application.

Changing the persistence locale does not change the language in which Decision Center displays rules. Changing it in Decision Center is necessary only to match the locale of Rule Designer when you synchronize your rule projects, and to access the tutorials in your locale.

Important: You must not change the persistence locale after you have saved a rule to the database.

Procedure

To set the persistence locale:

1. Enter a locale in the **Locale** field.
2. Click **Apply**.
3. Proceed as follows:
   - Click **Next** if you want to set the configuration parameters.
   - Click **Finish** if you do not want to change these settings.

Step 5: Set configuration parameters:

Many tasks that are related to customizing Decision Center require that you add or remove configuration parameters.

About this task

Decision Center uses the following configuration parameters to generate complete URLs in permalinks:

- **`teamserver.server.port`**: the port number
- **`teamserver.server.isSecure`**: `true` if the connection is secure
- **`teamserver.server.hostname`**: the name of the host.

To create, modify, or delete configuration parameters, you use the Set configuration parameters page in the Installation Manager wizard. You generate these parameters when you sign in to the Decision Center console for the first time after you have configured the database. You can use the Installation Settings wizard to change these parameters at any time.

The following table gives a description of the main configuration parameters available in `teamserver.war/WEB-INF/lib/teamserver-model-XXX.jar/ilog/rules/teamserver/preferences.properties`.

Note:

The parameters in the table include the **`teamserver`** prefix, which is not in the `preferences.properties` file. You must include the prefix when you set configuration parameters in the Installation Settings wizard.

Parameter	Used to
`teamserver.<extractorValidator>.class`	Specify a class of ruleset extractor validators to use for the `extractorValidator` name. The class must implement the `IlrExtractorValidator` interface. After you define this class, specify this name as the extractor validator to use when defining a ruleset extractor.

Parameter	Used to
`teamserver.build.path`	Define the location of the IRL cache in the file system. Compute the path as follows: • Use this property with the name of the user who started the server as the root for the cache (<*build.path*>_<*username*>). • If this property is not defined, use the system property **java.io.tmpdir** and add `rtscache`. For example, <*temp dir*>/rtscache_<*username*>). • If the system property is not defined, use the server directory and add `rtscache`. For example, <*server dir*>/rtscache_<*username*>).
`teamserver.brl.verbalizers`	Specify the list of locales for which a BAL verbalizer is defined.
`teamserver.brl.verbalizer.<locale>`	Specify the verbalizer class for the locale. The class must implement the `ilog.rules.vocabulary.verbalization.IlrVerbalizer` interface.

Procedure

1. Create a parameter or change an existing one.
   - To create a parameter, click **New**.
   - To change a parameter, select the check box next to the parameter and then click **Modify** to change the parameter or click **Delete** to remove it.
2. Click **Apply** to implement your changes.
3. Proceed as follows:
   - Click **Previous** if you want to make changes to previous settings.
   - Click **Finish**. The Installation log opens with a summary of the operations that you performed in the Installation Settings wizard.
4. Click **OK** to finish.

What to do next

You now have to sign in to the Decision Center console. Continue with the section Publishing a project to Decision Center.

Completing the configuration by using Ant tasks

Ant tasks provide an alternative method for completing or modifying the configuration. These tasks perform the same configuration steps as the Installation Settings wizard in the Decision Center console.

Setting up the environment for Ant tasks:

To run Decision Center Ant tasks, you must first set up the required environment variables.

Before you begin

Test your current version of Ant by typing the following command in a Windows Command Prompt or UNIX shell: **`ant -version`**

About this task

To run Ant tasks, you must have version 1.7.1 (or later) of Ant set up on your system. If Ant is not installed or your version is older than version 1.7.1, you must set up your environment to use the correct version of Ant. You can download Ant from the Apache web site, or you can use the Ant 1.7.1 distribution in `<ODM_InstallDir>/shared/tools/ant`.*<ODM_InstallDir>* is your Operational Decision Manager installation directory.

Communication between the Ant tasks and Decision Center supports the HTTP or HTTPS communication protocols. For more information, see Ant task communication protocol.

Note:

The **`appserver.name`** property configures the class path for the Ant tasks. If you need to add specific drivers to your class path, you can add them to the `<ODM_InstallDir>/teamserver/lib/classpath-teamserver.xml` file.

Procedure

To set up your environment to use Ant:

1. Set the ANT_HOME environment variable to `<ODM_InstallDir>/shared/tools/ant`.
2. Set the *JAVA_HOME* environment variable to the path to your JDK installation (1.6).
3. Add the directory `<ODM_InstallDir>/shared/tools/ant/bin` to your PATH environment variable.

   The Decision Center Ant tasks are defined in `<ODM_InstallDir>/teamserver/bin/build.xml` and executed by commands of the form:

   ```
   ant <taskName> <parameters list>
   ```

Results

To execute these Ant tasks, you must use the same Java virtual machine (JVM) version and vendor as the one that the application server uses.

Ant task parameters start with `-D`. Use them to set values such as the following ones:

- To specify the URL of the target application server:

  ```
  -Dserver.url=<server url>
  ```

- To specify the JNDI name of the data source to use for the task. The default value is `jdbc/ilogDataSource`.

  ```
  -DdatasourceName=<data source name>
  ```

Example

```
ant execute-schema -Dserver.url=<protocol://host:port>/teamserver/
 -DdatasourceName=jdbc/ilogDataSource -Dfile=my_sql_file.sql
```

The *<protocol://host:port>* URL is defined in the `<ODM_InstallDir>/teamserver/bin/teamserver-anttasks.properties` file. If your browser is not running on the same host as the application server, replace *localhost* with the address of the computer. If your web application is mapped to a host on a port that is different from the port number shown, change the port number to your host port number.

The `teamserver-anttasks.properties` file defines the values of some common parameters and other values that depend on the application server. If these parameters are properly defined in this file, you do not need to include them in your Ant task command. The content of the `teamserver-anttasks.properties` file is as follows:

```
# Default properties
# -----------------------------------
rtsAdmin.login=rtsAdmin
rtsAdmin.password=rtsAdmin1

protocol=http
server.host=localhost
server.port=8080
server.url=${protocol}://${server.host}:${server.port}/teamserver

datasourceName=jdbc/ilogDataSource

outputFile=output.sql

languagePackPath = .
languagePackOutputPath = ./generated

persistenceLocale =
selector =
branch =
override = false
```

Creating the database schema:

You can create the database schema in a single operation by using the **set-extensions** Ant task, or choose to create it step by step.

Creating schemas with the set-extensions Ant task:

For convenience, you can create the database schema by using the **set-extensions** Ant task.

About this task

Extensions to the Decision Center rule model are stored in two XML files.
- Model description: This file usually has the file name extension `.brmx`.
- Initialization of enumerations and hierarchies: This file usually has file name extension `.brdx`.

You can use Ant tasks to load the rule model from the two XML files and build the SQL script that is necessary to get the proper database schema.

Note: To run these Ant tasks, you must use the same Java virtual machine version and vendor as the one used by the application server.

Alternatively, you can create the database schema step by step, which is useful if you want to look at the generated SQL schema. See "Creating the schema using a step-by-step sequence" on page 41.

Procedure

Run the **set-extensions** Ant task.
This Ant target runs **gen-create-schema** + **execute-schema** + **upload-extensions** + **upload-roles**. Set the following parameters:

-Dserver.url=<*server url*>

-DdatasourceName=<*data source name*>

-DextensionModel=<*model file*>
The model description file, with the `.brmx` extension

-DextensionData=<*data file*>
The model data description, with `.brdx` extension

[-DdbSchemaName=<*database schema name*>**]**
You can use this optional parameter to specify the database schema name. Otherwise, Decision Center uses the database user name as the schema name. However, some databases allow some users to access several schemas, and the default schema does not always reflect the user name.

[-Droles=<*role list*>**]**
You can use this optional parameter to upload the list of roles to Decision Center. This list is specified as `"role1 role2"`. For example:

```
ant upload-roles  -Droles="rtsUser rtsConfigManager Eligibility Validator"
```

Creating the schema using a step-by-step sequence:

If you want to look at the generated SQL schema, you can create it step-by-step.

Creating the database schema script:

You can create the database schema script by using the **gen-create-schema** Ant task.

Procedure

To create the SQL script that is necessary to create or update the database schema, run the **gen-create-schema** Ant task with these parameters:

-Dserver.url=<*server url*>

-DdatasourceName=<*data source name*>

-DextensionModel=<*model file*>
The model description (`.brmx` extension).

-DextensionData=<*data file*>
The model data description (`.brdx` extension).

[-DdbSchemaName=<*database schema name*>**]**
You can use this optional parameter to specify the database schema name in which the Decision Center tables are stored. If you do not specify the parameter, Decision Center uses the database user name as the schema name. However, some databases allow some users to access several schemas, and the default schema is not always named the same as the user.

[-DoutputFile=<SQL file>]
The name of the file that stores the generated SQL script. If this parameter is not specified, the task creates a file named `output.sql` in the directory that is defined as `basedir` in the `build.xml` file.

```
ant gen-create-schema  -DextensionModel=my_model_file.brmx -DextensionData=my_data_file.brdx -Dout
```

Results

1. The task connects to the specified data source from the application server.

2. The task checks whether this data source points to an existing Decision Center database.
   - If a database does not exist, the task builds the SQL script to create a fresh database schema to store the model.
   - If a database does exist, the task builds the SQL script that is necessary to update the existing database schema.

Executing the database schema script:

You execute the database schema script.

Procedure

To execute the SQL script that you created, run the **execute-schema** Ant task with these parameters:

-Dserver.url=<*server url*>

-DdatasourceName=<*data source name*>

[-Dfile=<*SQL file*>**]**
: The name of the file to execute, which corresponds to the script that you created. If you do not specify this parameter, the task attempts to execute a file named `output.sql` in the directory that is defined as `basedir` in the `build.xml` file.

```
ant execute-schema  -Dfile=my_sql_file.sql
```

Uploading the database schema extension:

You upload the database schema extension.

Procedure

To store the rule model description in the database schema, run the **upload-extensions** Ant task with these parameters:

-Dserver.url=<*server url*>

-DdatasourceName=<*data source name*>

-DextensionModel=<*model file*>
: The model description (`.brmx` extension).

-DextensionData=<*data file*>
: The model data description (`.brdx` extension).

```
ant upload-extensions  -DextensionModel=my_model_file.brmx -DextensionData=my_data_file.brdx
```

Results

The description is stored in the database so that Decision Center applications can load it when they start. It is also used by **gen-create-schema** to get the current model description to run a `diff` with the new schema.

In a cluster, you must restart the servers and close all current sessions.

Uploading a list of roles or groups to the database:

In addition to creating groups in your application server when you set up security access, you must upload groups to the database.

Before you begin

1. Add all the groups that you want to see in the available list when you enforce project security or setting permissions in Decision Center.
2. Create the default groups for `rtsUser` and `rtsConfigManager`, and upload your custom groups.

About this task

You must upload roles or groups only if you want to use the Decision Center project access and permissions mechanisms. For more information, see the topics on Groups and Permissions in the Decision Center online help.

You do not have to upload the `rtsAdministrator` group or the `rtsInstaller` group. The Administrator group has access to everything and an Installer user must belong to another group.

Note: To run these Ant tasks, you must use the same Java virtual machine version and vendor as the one used by the application server.

Procedure

To store in the database the list of roles or groups to be used by the application, run the **upload-roles** Ant task with the following parameters:

-Dserver.url=<*server url*>

-DdatasourceName=<*data source name*>

-Droles=<*role list*>
<*role list*> is the list of roles or groups to upload to Decision Center, specified as "group1 group2".

```
ant upload-roles  -Droles="rtsUser rtsConfigManager Eligibility Validator"
```

Removing a database schema:

You can create an SQL script to remove (drop) a database schema by using the **gen-drop-schema** Ant task.

About this task

To remove a database schema, you proceed in two steps:

1. Create the SQL script that is necessary to remove the database schema.
2. Execute the SQL script that you created.

Procedure

1. To create the SQL script to delete a database schema, run the **gen-drop-schema** Ant task with the following parameters:

   -Dserver.url=<*server url*>

   -DdatasourceName=<*data source name*>

   -DextensionModel=<*model file*>
   The description of the database schema to remove.

   [-DdbSchemaName=<*database schema name*>**]**
   You can use an optional parameter to specify the database schema name. If you do not specify this parameter, Decision Center uses the database user

name as the schema name. However, in some databases, users can access several schemas and the default schema is not always named as the user.

[-DoutputFile=<*SQL file*>**]**
The name of the file that stores the generated SQL script. If you do not specify this parameter, the task creates a file named `output.sql` in the directory that is defined as `basedir` in the `build.xml` file.

```
ant gen-drop-schema  -DextensionModel=my_model_file.brmx -DoutputFile=my_sql_file.sql
```

2. To execute the SQL script that you created, run the **execute-schema** Ant task with these parameters:

   -Dserver.url=<*server url*>

   -DdatasourceName=<*data source name*>

   [-Dfile=<*SQL file*>**]**
   The name of the file to execute, which corresponds to the script that you created. If you do not specify this parameter, the task attempts to execute a file named `output.sql` in the directory that is defined as `basedir` in the `build.xml` file.

   ```
   ant execute-schema  -Dfile=my_sql_file.sql
   ```

   The task connects to the specified data source from the application server. It reads the model description that is passed in the parameters, and generates the SQL script to remove the existing schema. Because many database tables are linked through foreign keys, these tables must be removed in a specific order and the script generation handles these constraints.

Example

Here is the complete code sample:

```
ant gen-drop-schema  -DextensionModel=my_model_file.brmx -DoutputFile=my_sql_file.sql
ant execute-schema  -Dfile=my_sql_file.sql
```

Defining and uploading message files:

You can define and upload message files to Decision Center by using the **upload-messages** Ant task.

Message files contain the display text that is associated with the extensions to the rule model that is contained in the `.brmx` and `.brdx` files. For example:

```
status=Status
effectiveDate=Effective Date
expirationDate=Expiration Date
new=New
defined=Defined
```

The default messages file is provided in: <*ODM_InstallDir*>`/teamserver/bin/defaultextensionmessages_<LOCALE>.properties`

Note: The contents of the messages files must conform to the ISO-LATIN-1 standard.

You must have a messages file for each locale that you use. Upload the messages file to Decision Center by running the **upload-messages** Ant task with these parameters:

- **-Dserver.url=<server url>**
- **-DdatasourceName=<data source name>**

- **-Dlocale=<locale>**
- **-DmessageFile=<message file>**

```
ant upload-messages  -Dlocale=en_US -DmessageFile=mymessages.properties
```

Setting the persistence locale:

The persistence locale is used to determine the language in which rules are stored in the Decision Center database.

About this task

The persistence locale is set when you deploy the Decision Center archive to your application server, which means that the rules in the database are stored in the locale of the Decision Center application.

Changing the persistence locale does not change the language in which rules display in Decision Center. Changing the persistence locale in Decision Center is necessary only to match the locale of Rule Designer when you synchronize your rule projects, and to access the tutorials in your locale.

Important: You must not change the persistence locale after you have saved a rule to the database.

Procedure

To set the persistence locale by running an Ant task:

1. Open the `<`*`ODM_InstallDir`*`>/teamserver/bin/teamserver-anttasks.properties` file.

   This file defines the value of some common parameters.
2. Add your locale to the **`persistenceLocale`** property and save the `teamserver-anttasks.properties` file.

   For example: `persistenceLocale = fr_FR`
3. Run the Ant task in this form: `ant` *`taskName parameters_list`*

   Alternatively, you can add the parameter to the command line. For example: `ant` *`taskName`* `-DpersistenceLocale=fr_FR`

Adding or removing configuration parameters:

For many tasks that are related to customizing Decision Center, you must add or remove configuration parameters.

The following configuration parameters, used to generate complete URLs in permalinks, are generated the first time you sign in to Decision Center after the database is configured. You can use the Installation Settings wizard to set these parameters beforehand or change them afterward:

- **`teamserver.server.port`**: The port number
- **`teamserver.server.isSecure`**: `true` if the connection is secure.
- **`teamserver.server.hostname`**: The name of the host

The following table gives a description of the main configuration parameters available in `teamserver.war/WEB-INF/lib/teamserver-model-XXX.jar/ilog/rules/teamserver/preferences.properties`.

Parameter	Use
teamserver.<extractorValidator>.class	Specify a ruleset extractor validator class to use for the extractorValidator name. The class must implement the IlrExtractorValidator interface. After this class is defined, specify this name as the extractor validator to use when defining a ruleset extractor.
teamserver.build.path	Define where the cache of the IRL is located on the file system. The path is computed as follows: 1. First, use this property with the name of the user who started the server as the root for the cache: *<build.path>_<username>*. 2. If the path is not defined, use the system property **java.io.tmpdir** and add rtscache. For example, *<temp_dir>*/rtscache_*<username>*. 3. If the system property is not defined, use the server directory and add rtscache. For example, *<server_dir>*/ rtscache_*<username>*.
teamserver.brl.verbalizers	Specify the list of locales for which a BAL verbalizer is defined.
teamserver.brl.verbalizer.<locale>	Specify the verbalizer class for the specified locale. The class must implement the IlrVerbalizer interface.

You can use the following Ant tasks to add or remove configuration parameters:

set-config-param

Sets a configuration parameter for a specified user. If the user is not specified, it sets a *global parameter*.

Parameters:

- **-Dserver.url=<server url>**
- **-DdatasourceName=<data source name>**
- **[-Duser=<username>]**
- **-Dkey=<parameter key>**
- **-Dvalue=<parameter value>**

For example:

```
ant set-config-param  -Dkey=locale -Dvalue=en_US
```

remove-config-param

Drops the configuration parameter for a specified user. If the user is not specified, it drops the global configuration parameter.

Parameters:

- **-Dserver.url=<server url>**
- **-DdatasourceName=<data source name>**
- **[-Duser=<username>]**
- **-Dkey=<parameter key>**

print-config-param

Prints the global parameters or specified user parameters if the *username* value is specified. If no key is specified, all keys are printed.

Parameters:

- **-Dserver.url=<server url>**
- **-DdatasourceName=<data source name>**
- **[-Duser=<username>]**
- **-Dkey=<parameter key>**

Repackaging the Decision Center archive:

You can repackage the Decision Center archive by using an Ant task.

When you add new `.jar` files to the Decision Center archive, you must repackage the archive by running the **repackage-ear** or **repackage-war** Ant task. This task does not use the **server.url** and **datasourceName** parameters.

If you have customized Decision Center, you must package the custom `.jar` files before you use the Ant task to repackage the Decision Center.

The **repackage-ear** or **repackage-war** Ant task takes the following parameters:

-DtargetEar=<target ear>
: **-DtargetWar=<target war>** for the Tomcat application server

-DsourceEar=<source ear>
: **DsourceWar=<source war>** for the Tomcat application server

-DdescriptorsDir=<descriptors directory>
: A directory that is copied into the `META-INF` directory of the target EAR (not mandatory).

-DadditionalJars=<"myjar1.jar,myjar2.jar, myjarn.jar">
: Additional `.jar` files to store in the `lib` directory of the target archive (not mandatory).

-DtmpDir=<directory>
: A directory that you can specify to store temporary files (not mandatory).

-DwebResourcesDir=<web resources directory>
: A directory that is copied into the WAR library (not mandatory).

-Dconsole=both|enterprise|business
: Specifies whether to repackage the Business or the Enterprise WAR files. The default is both.

Configuring the search function of the Decision Center Business console

The search function in the Decision Center Business console is based on the Solr search engine. You must configure the engine to provide this functionality.

Using a remote Solr search engine

You can set up the search function in Decision Center Business console to work with a remote instance of the Apache Solr search engine.

About this task

The search function in Decision Center Business console uses an embedded instance of the Apache Solr search engine. Alternatively, you can have the search function work with a remote instance of the search engine, which you can run on another computer, or the same computer but in a dedicated web application.

Procedure

To configure the search function to run with a remote instance of the Solr search engine:

1. Install the Apache Solr search engine on another computer, or as part of a dedicated web application on your computer.

   For information on installing the Solr server, visit the Apache Solr website.
2. Locate the `decisioncenter-solr-home.zip` file in the Decision Center `teamserver` folder on your computer.

   The Decision Center installation program placed the folder on your computer.
3. Decompress the file in a directory in the remote instance of the Solr server.
4. Configure the home directory of the remote Solr server to use the location of the decompressed configuration files.

   For more information, see the documentation on the Apache Solr website.
5. Configure the Decision Center `preferences.properties` file to point to the URL of the Solr server.

Setting parameters for the Solr search engine

You can configure the search function in the Decision Center Business console to run with the Apache Solr search engine.

Decision Center Business console provides a function for searching rule projects. By default, it uses an embedded instance of the Apache Solr search engine.

Alternatively, however, you can choose to run the search function with a remote instance of the search engine, which you can install on another computer, or on the same computer but in a dedicated web application. For more information, see "Using a remote Solr search engine" on page 47.

You can set three parameters for the search engine:

Table 5. Search parameters

Parameter	Description
search.SearchProvider	This parameter takes one of the following values: • `SolrEmbedded`: Use this value to select the embedded Solr search engine. • `SolrRemote`: Use this value to select a remote instance of the Solr search engine.
search.SolrEmbeddedDataDir	Use this optional parameter to direct the index of the embedded version (`SolrEmbedded`) to a specific directory on the Decision Center.
search.SolrRemoteUrl	Use this parameter with `SolrRemote` to provide the URL of the remote Solr search engine.

You set the configuration parameters for the search engine within the `preferences.properties` file for Decision Center. For more information on the `preferences.properties` file, see Configuration parameter maintenance.

The following table provides examples for setting the parameters in `preferences.properties` to work with the Solr search engine:

Table 6. Configuration table for search properties

Solr server	Parameter settings

External server
 	
The URL of the external server depends on the installation. For this example, the remote address of the search engine is `http://mysearchserver:8983/solr`.
	Set the `preferences.properties` file as follows:
	
	`search.SearchProvider=SolrRemote`
	`search.SolrRemoteUrl=http://mysearchserver:8983/solr`

Embedded serverIn this example, you store the index in the `c:/temp/DC-SearchIndex` directory on your computer.	Set the `preferences.properties` file as follows:
	
	`search.SearchProvider=SolrEmbedded`
	`search.SolrEmbeddedDataDir=c:/temp/DC-SearchIndex`

If you do not specify a directory for **`search.SolrEmbeddedDataDir`**, Decision Center stores the search index in a temporary directory. When the server stops running, it also stops using the temporary directory. When the server restarts, it creates a new temporary directory, and completely re-indexes the repository.

If you specify a directory for **`search.SolrEmbeddedDataDir`**, the directory and its content persist across server restarts, and the server does not re-index the repository with each restart.

Additional steps to configure Decision Validation Services

To configure Decision Validation Services, additional steps are necessary.

Before you start

Before you can deploy and configure Decision Validation Services, you must install Rule Execution Server and its console. A table summarizes the sequence of steps that ensure a successful configuration.

Make sure that the following software is installed:

1. Install Rule Execution Server.
2. Install Decision Center.
3. Configure both of these applications on the same application server.

Restriction:

Only the users who belong to the `resAdministrators` or `resMonitors` groups can see the Decision Warehouse tab in the Rule Execution Server console after Decision Validation Services is configured.

What steps to follow

The following table summarizes the configuration steps for Decision Validation Services.

Step	Mandatory/Optional
"Step 1: Creating Decision Warehouse database resources"	Optional. If you use the Rule Execution Server console to create database resources, you do not have to perform this task. For more information, see Creating a database schema from the Rule Execution Server console.
"Step 2: Deploying the Decision Validation Services archive" on page 52 This archive is the default Scenario Service Provider (SSP) archive that is packaged with the Decision Validation Services installer.	Optional. Do this step if you want to check the availability of the feature **Note:** You can also use the default SSP archive if you have an XML XOM. In this case, you do not have to repackage the archive to include the XML XOM.
"Step 3: Checking the availability of Decision Validation Services" on page 52	Optional
"Step 4: Packaging Decision Validation Services by using Ant" on page 54	Optional. You can use this task as an alternative method for configuring or modifying the Decision Validation Services installation.

Step 1: Creating Decision Warehouse database resources

You can use SQL scripts to create a dedicated schema in the database.

If you set the Rule Execution Server persistence to `datasource` or `jdbc` and you intend to use Decision Warehouse to store your test execution results, you must create a dedicated schema in the database that contains these tables and views. To do so, you can use the provided SQL scripts, located in `<ODM_InstallDir>/executionserver/databases`. A `readme` file in this directory provides additional information about the scripts.

The script that creates the Decision Warehouse database schema is named `trace_<database_name>.sql`.

Note:

The Installation Settings wizard in the Rule Execution Server console creates all the necessary tables for Rule Execution Server and for Decision Warehouse. If you are configuring Decision Validation Services and you have already run the Installation Settings wizard to create the tables, you do not have to create database resources manually. However, if you did not use the wizard to create database resources, you must run the script to create the Decision Warehouse database schema.

When you use DB2 (except on z/OS), the scripts that create the Rule Execution Server database tables are written for databases that use automatic storage.

- BP32K is the buffer pool that is expected in SYSCAT.BUFFERPOOLS. If BP32K is not there, you can use the existing buffer pool or create a new buffer pool named BP32K. Use the following command to query SYSCAT.BUFFERPOOLS for the existing buffer pool:

  ```
  Select * from SYSCAT.BUFFERPOOLS
  ```

  Otherwise, use the following command to create a buffer pool named BP32K:

  ```
  CREATE BUFFERPOOL BP32K SIZE 2000 PAGESIZE 32K
  ```

- You must update the trace_db2.sql script and select the custom option in the Installation Settings wizard to run it. Modify the following line in the script to specify storage for the table space:

  ```
  CREATE TABLESPACE RESDWTS PAGESIZE 32K BUFFERPOOL BP32K;
  ```

  Here is an example of the table space specification in the script:

  ```
  CREATE TABLESPACE RESDWTS PAGESIZE 32K  MANAGED BY Database
  USING [ FILE 'C:\DB2\Container.file' 640 ] BUFFERPOOL  BP32K;
  ```

- You might have to further modify the script based on your database settings.

Note:

If you completed this task as part of the Rule Execution Server configuration, you do not have to do this a second time. It is included here solely for completeness.

If you use Command Editor to run the scripts, you must log in with the credentials that you use for the data source for Rule Execution Server.

Use any tool that can handle SQL to import and run the SQL scripts. The tools provided for each database include:

Database	**Database tool**
IBM DB2	DB2 command line processor
Derby	ij command line processor
MySQL	mysql command line processor
Oracle	sqlplus command line processor
Postgre SQL	Postgre SQL command line tool
SQL Server	Query Tool
Sybase	isql command line processor

To access the database, the database user must have the following credentials:
- A user ID and a password
- Complete privileges on the tables and view of the schema (create, insert, delete)
- create index privileges
- On Oracle, create trigger and create sequence privileges. If you use an Oracle database, run all the scripts in the SQL Plus client.

Install a database client for the database that you use. Refer to the documentation of the database for more information.

The default CLOB size might not be sufficient for the FULL_EXECUTION_TRACE field in the EXECUTION_TRACES table. You might need a size qualifier if SQL raises exceptions with the <*Lob-Value*> reason code.

Step 2: Deploying the Decision Validation Services archive

After you have Decision Warehouse database resources available, you deploy the Decision Validation Services archive.

Before you begin

To deploy a Decision Validation Services archive, you work in the WebSphere Integrated Solutions Console. Log in to the console as explained in "Before you start: Opening the administration console" on page 23.

About this task

If you want to check the availability of Decision Validation Services, you must deploy the default EAR file for the Scenario Service Provider (SSP). You follow the same procedure whether you deploy the default SSP archive that is packaged with the installer or any subsequently repackaged archive. If you have an XML XOM, you can also use the SSP EAR file to test your rules. In this case, because the ruleset archive inside a RuleApp includes an XML XOM, you do not have to repackage the SSP EAR file to include the XOM.

Important: You must deploy the Decision Validation Services archive on the same server as the XU.

Procedure

To deploy the archive:

1. At the bottom of the page, click **Deployments**.
2. In the Summary of Deployments page, under **Deployments**, click **Install**.
3. In the Install Application Assistant, click **upload your file(s)**.
4. Click **Browse** next to **Deployment Archive**, upload one of the following files, and click **Next**.
   - `<`*`InstallDir`*`>/executionserver/applicationservers/WebLogic10/jrules-ssp-WL10.ear`
   - `<`*`InstallDir`*`>/executionserver/applicationservers/WebLogic12/jrules-ssp-WL12.ear`
5. Select the EAR file and click **Next**.
6. Select **Install this deployment as an application** and click **Next**.
7. Select **Use the defaults defined by the deployment's targets** and click **Finish**.
8. In the Summary of Deployments page, select the EAR file and select **Start** > **Servicing all requests**.
9. In the Start Application Assistant, click **Yes** to start the deployment.
10. Verify that the archives are deployed.

    The **State** option must be set to **Active**.

Step 3: Checking the availability of Decision Validation Services

A specific testing URL is designed for you to check the rule session type, the DAO factory class for trace persistence, the class used to cache the Decision Validation Services job, and the pool size for asynchronous execution.

To check the availability of Decision Validation Services:

1. Enter the URL `http://<host>:<port>/testing` in a web browser.

2. Log in to the (Scenario Service Provider) SSP application in one of the Rule Execution Server roles.

   The application displays a home page, which contains information about the SSP server.

Version

The version of Decision Server used.

Patch level

The patch level of Decision Server used.

License information

The type of license of this version.

RuleSession

The rule session type (`POJO` or `J2SE`).

DAO Factory Class

The DAO (Data Access Object) factory class that is used to persist the trace into the data warehouse.

Job store class

The name of the class that is used to persist the Decision Validation Services job into a cache to free the memory during long computations.

Job pool size

The size of the pool for asynchronous execution.

Started since

The time and date when the SSP started.

Jobs currently running

The About screen provides information about the jobs that are currently running after you run Decision Validation Services in Decision Center:

- A **Job ID** is listed in the table when a user clicks **Run** in Decision Center.
- The **Created** column records the date and time when each job is initialized.
- The **Status** column shows the number of scenarios that have already been tested, compared to the total number of scenarios.
- The **Start time** records the time when a resource is allocated for the job.
- The **Parts** column records the number of parts in the job:
  - A job that is not executed in parallel has one part.
  - A job that is executed in parallel has one or more parts.
- The **End time** records the time when the execution of the job is complete, that is, all of the scenarios in the job have been tested.

The report for the job is automatically downloaded by Decision Center at the end of the execution. If the scenario suite is run in the background, the user downloads the report by viewing the list of scenario suites, and then clicking the report link when it becomes available. After the report is viewed, the job is removed from the table. The job remains in the table until the report is downloaded.

Step 4: Packaging Decision Validation Services by using Ant

You can configure Decision Validation Services archives by using an Ant task.

About this task

This section is for users who install the product on Windows and other supported distributed platforms only.

You can configure Decision Validation Services archives by running the **ssp-setup** Ant task.

Procedure

1. Define the Ant task in your build file by using the <*taskdef*> Ant element in one of the following ways:
   - Define the task at the top level, or within a specific target.
     ```
     <taskdef resource="res-tasks.properties"
     classpath="${<InstallDir>}/executionserver/lib/jrules-res-setup.jar"/>
     ```
   - If the JAR file is available in your system, you can write the following code:
     ```
     <taskdef resource="res-tasks.properties"/>
     ```
2. Use the **ssp-setup** Ant task to update a Scenario Service Provider (SSP) artifact to your specific configuration and XOM.
3. Run the Ant task in one of the following ways:
   - From the command line: Execute Ant in the appropriate directory, followed by the name of the build file if necessary.
   - From Eclipse: Right-click the Ant file and click **Run**.

What to do next

You can now deploy the testing archive.

Allowing non-administrators to run diagnostics

After you have created users and added them to groups, you might want to allow non-administrators to run diagnostics in Rule Execution Server.

Before you begin

You work in the WebLogic Server Administration Console. See “Before you start: Opening the administration console” on page 1 for details.

About this task

By default, the `resDeployer` and `resMonitor` users do not have WebLogic Server administrator rights, which they need to run the diagnostics. The following procedure shows how to add a Java Management Extensions (JMX) policy for these users to access all the MBeans. For the **JMX Policy Editor** to be available on the Policies page, you must first activate a specific option to protect JMX access.

Procedure

1. From the breadcrumbs at the top of the console, or from the Home Page, open the Summary of Security Realms page, and in the Realms table, click **myrealm**.
2. On the Settings for myrealm page, on the **Configurations** > **General** tab, select the **Use Authorization Providers to Protect JMX Access** check box.

3. Click **Save** and restart the server.

   Tip: It is not necessary to restart the Administration Console.
4. After the server is restarted, click the **Roles and Policies** tab, and then the **Realm Policies** tab.
5. Under **Policies**, click **JMX Policy Editor**.
6. On the JMX Policy Editor page, check that **GLOBAL SCOPE** is selected and click **Next**.
7. In the next panel, check that **ALL MBEAN TYPES** is selected and click **Next**.
8. In the Attributes and Operations table, select **Operations: Permission to Invoke** and click **Create Policy**.
9. Click **Add Conditions**, select **Role** from the **Predicate List**, and click **Next**.
10. Type `Admin` in the **Role Argument Name** field and click **Add**, and then click **Finish**.
11. On the Edit JMX Policies page, click **Add Conditions**, select **Group** from the **Predicate List** list, and click **Next**.
12. Add the groups that do not have administrator privilege and whom you allow to run the diagnostic tests. For example, to add all the `resDeployers` and `resMonitors` groups follow these steps:
    a. Type `resDeployers` in the **Group Argument Name** field and click **Add**.
    b. Type `resMonitors` in the **Group Argument Name** field and click **Add**.
    c. Click **Finish**

    The policy conditions and the overridden policy are shown:
    ```
    Group: resDeployers or resMonitors

    Role: Admin
    ```
13. Click **Save**.
14. Restart WebLogic Server.

Chapter 2. Verifying your configuration of Decision Center

You can verify that you have correctly configured Decision Center by publishing some projects, opening the consoles, and running the diagnostics.

Publishing a project to Decision Center

After completing the configuration, Decision Center is ready to be used but does not contain any rule projects. You publish rule projects from Rule Designer.

About this task

To publish a rule project to Decision Center, the project must be imported into your workspace.

The procedure uses the Decision Center tutorial projects as an example of how to import and publish a rule project. If you want to carry out the Decision Center tutorials, you have to publish the following projects:

- `loanvalidation-rules` (with `loanvalidation-xom`)
- `loanvalidation-rules-dependent`
- `squery-loanvalidation-rules` (with `squery-loanvalidation-xom`)

Procedure

1. To open Rule Designer, click **Start** > **All Programs** > **IBM** > ***package_group*** > **Rule Designer**.
2. In Rule Designer click **File** > **Import** > **General** > **Existing Projects into Workspace**, and click **Next**.
3. Click **Select root directory**, browse to `<InstallDir>/studio/tutorials/shared`, and click **OK**.
4. Select the projects and click **Finish**.
5. Right-click the `loanvalidation-rules` rule project, and click **Decision Center** > **Connect**.
6. Complete the Decision Center Configuration dialog as follows. The warning message `Connection not established` displays until you establish the connection.

   User name
   rtsAdmin

   Password
   rtsAdmin

   URL `http://localhost:<port>/teamserver`

   Data source
   Leave this field empty.

   Note: If security is enabled, use `https://localhost:<PORT_NUMBER>/teamserver`
7. Click **Connect**.

   The connection is established when the warning message closes and the **Project configuration** area becomes active.

8. In the **Project configuration** area, check that **Create a new project on Decision Center** is selected, and then click **Finish**.
9. The Synchronize Complete - Decision Center Participant dialog opens when the publishing process is complete. Click **OK** to close this dialog.
10. A dialog opens asking you if you want to change to Team Synchronizing perspective. Click **Yes**.

    An empty Synchronize view opens, indicating that the projects in Rule Designer and Decision Center are the same. This means that your rules are now published to Decision Center.
11. Repeat for the other rule projects.

What to do next

You can now open the Decision Center Enterprise console and perform diagnostics.

Opening the Decision Center consoles

After you have deployed the Decision Center EAR or WAR to your application server, you can open the Decision Center consoles.

You can open the consoles by using the following URLs in a web browser:

- **Enterprise console**: `http://localhost:<PORT_NUMBER>/teamserver`
- **Business console**: `http://localhost:<PORT_NUMBER>/decisioncenter`

Note: If your browser is not running on the same host as the application server, replace `localhost` with the address of the machine. If your web application is mapped to a host with a port that is different from the default port, use the port number of the host.

By default, the data source is `jdbc/ilogDataSource`. If you want to specify a different data source, you have to pass it as a request parameter in the URL. For example:

`http://localhost:7001/teamserver?datasource=jdbc/serverextendedbrm`.

The locale of the sign-in page is English by default. You can specify a locale parameter in the URL that switches the sign-in page to the required locale. For example:

`http://localhost:<port>/teamserver?locale=es` (assuming that your message files are localized).

If you sign in with another locale in the URL and want to change the locale afterward, click **Options** in the top banner of the Enterprise console or **Profile** in the Business console. This saves the locale and restores it the next time you sign in.

If you open Decision Center but no database exists, you automatically access the Installation Settings wizard with only the **Install** tab available.

After completing the installation, Decision Center is ready to use but does not contain a rule project. You have to publish a rule project from Rule Designer.

A diagnostics tool, available in the Configure tab of the Enterprise Console, shows a report on the status of your Decision Center configuration.

To learn more about Decision Center, see Decision Center.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM United Kingdom Laboratories,
Mail Point 151,
Hursley Park,
Winchester,
Hampshire,
England SO21 2JN

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM United Kingdom Laboratories,
Mail Point 151,
Hursley Park,
Winchester,
Hampshire,
England SO21 2JN

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Index

Printed in USA