

A cluster of colorful gears in various sizes and colors (green, blue, yellow, red, purple) is located in the top-left corner of the page.

**betaWorks**

## IBM Integration Bus

# Implementing Business Rules with IBM Operational Decision Management using the Employee Service scenario

Featuring:

- Creating business rules in the ODM Rule Designer
- Deploying to the ODM Decision Center and Rule Execution Server
- Using the rule in the IIB Toolkit with a Decision Service node
- Testing the rule with IIB
- Updating the rule with the ODM Business Console

**June 2015**

Hands-on lab built at product  
version 10.0.0.0

<b>1. INTRODUCTION TO DECISION SERVICES.....</b>	<b>3</b>
1.1 SCENARIO SCHEMATIC .....	3
<b>2. DEFINE THE BUSINESS RULE IN ODM .....</b>	<b>4</b>
2.1 CREATE THE RULE IN THE RULE DESIGNER.....	4
2.2 EXPORT THE RULEAPP ARCHIVE .....	16
2.3 ADD THE ODM RULE TO THE ODM DECISION CENTER REPOSITORY .....	19
2.3.1 <i>Step 1 – Add the rule into the Decision Center repository</i> .....	20
2.3.2 <i>Step 2 – Deploy the rule from the Decision Center to the Rule Execution Server</i> .....	24
<b>3. CONFIGURE THE INTEGRATION BUS APPLICATION WITH THE ODM RULE .....</b>	<b>34</b>
3.1 IMPORT THE RULE INTO THE INTEGRATION BUS TOOLKIT.....	34
3.1.1 <i>Review the Integration Bus Configurable Services for Decision Services</i> .....	45
<b>4. TEST THE SERVICE WITH THE DEPLOYED RULE.....</b>	<b>48</b>
4.1 TEST THE SERVICE WITH THE IIB FLOW EXERCISER .....	48
4.2 TEST THE SERVICE WITH SOAPUI (OPTIONAL) .....	52
<b>5. UPDATE THE BUSINESS RULE .....</b>	<b>54</b>
5.1 UPDATE THE RULE IN THE BUSINESS CONSOLE .....	54
5.2 USE THE ODM ENTERPRISE CONSOLE TO DEPLOY THE UPDATED RULE .....	60
<b>6. TEST THE SERVICE AGAIN.....</b>	<b>62</b>

# 1. Introduction to Decision Services

You can use the IBM Integration Bus Toolkit to write business rules by using natural language, so that they can be read easily by business users (for example, a business analyst). In the Integration Toolkit, you create a decision service, which is a collection of rules that are used to process a message.

A DecisionService node executes those business rules to provide operations like routing, validation, and transformation. You can either write rules in the Integration Toolkit, or import rules from IBM Operational Decision Manager. You can also retrieve rules from an external IBM Operational Decision Manager repository.

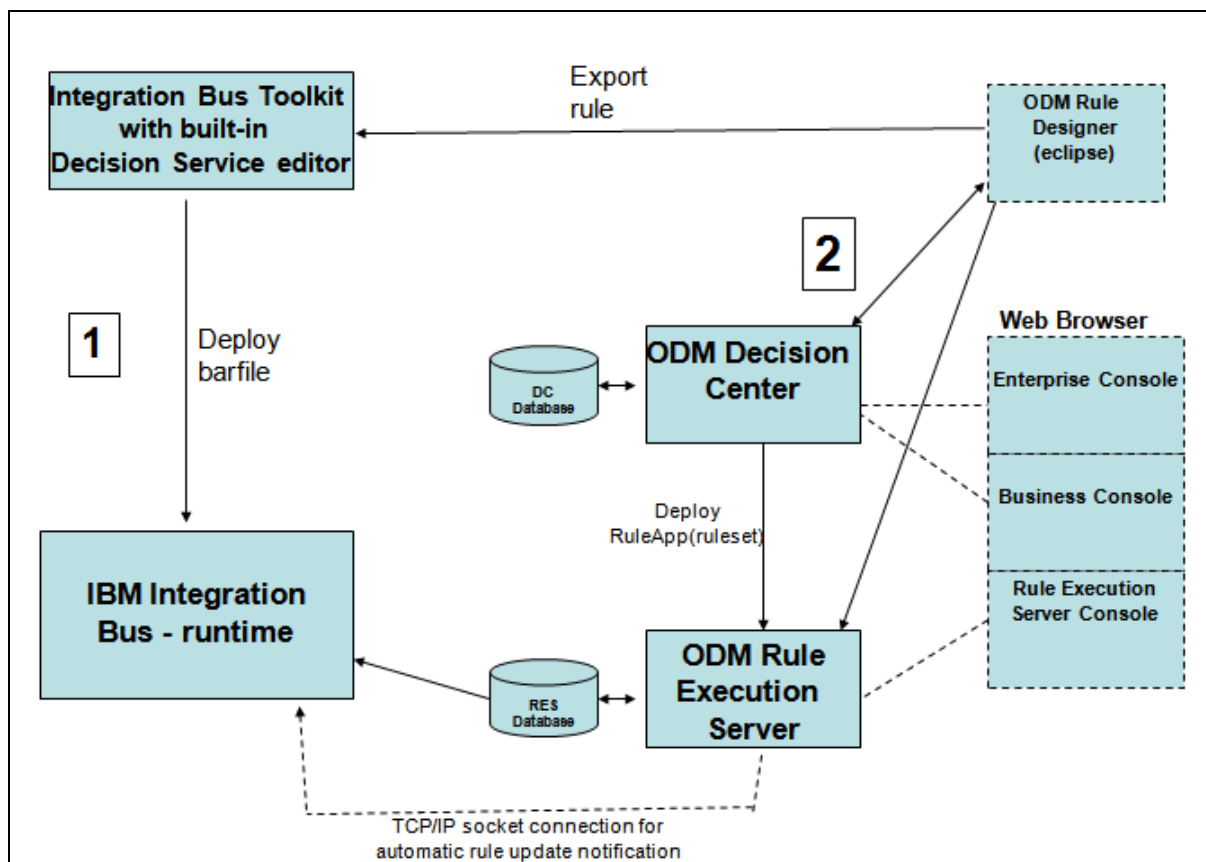
## 1.1 Scenario Schematic

The following schematic shows the primary components of the Integration Bus environment when used in conjunction with IBM Operational Decision Management.

On the left side, (1), the embedded ODM components enable IBM Integration Bus to process rules defined in the Integration Bus Toolkit. No components from the IBM ODM product are required for this.

On the right side, (2), shows the primary ODM components that are needed to define rules within the ODM system, and to deploy and load those rules into the Integration Bus. The Integration Bus runtime loads rules directly from the ODM database. Updates to those rules can be automatically detected and loaded by the Integration Bus runtime.

This lab will focus primarily on this scenario (2), where the business rule is defined and managed in ODM.



## 2. Define the Business Rule in ODM

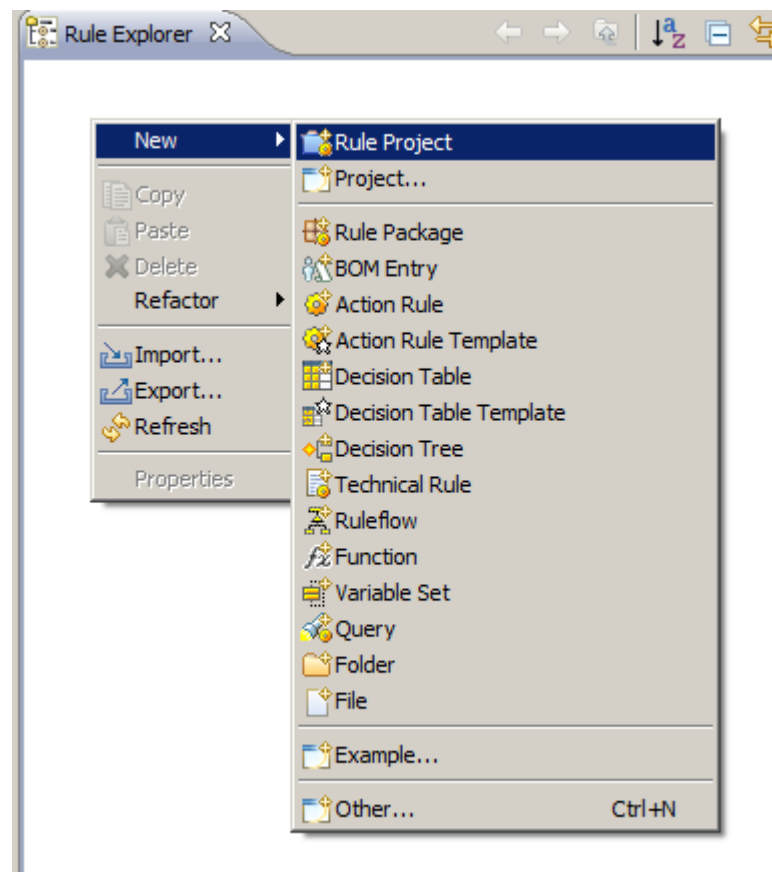
### 2.1 Create the Rule in the Rule Designer

This section will show you how to define rules using the IBM Operational Decision Manager Rules Designer, how to store these in the ODM repository (Decision Center), how to update them with the Business Console, how to deploy them to the Rule Execution Server, and how to use these rules from an Integration Bus application.

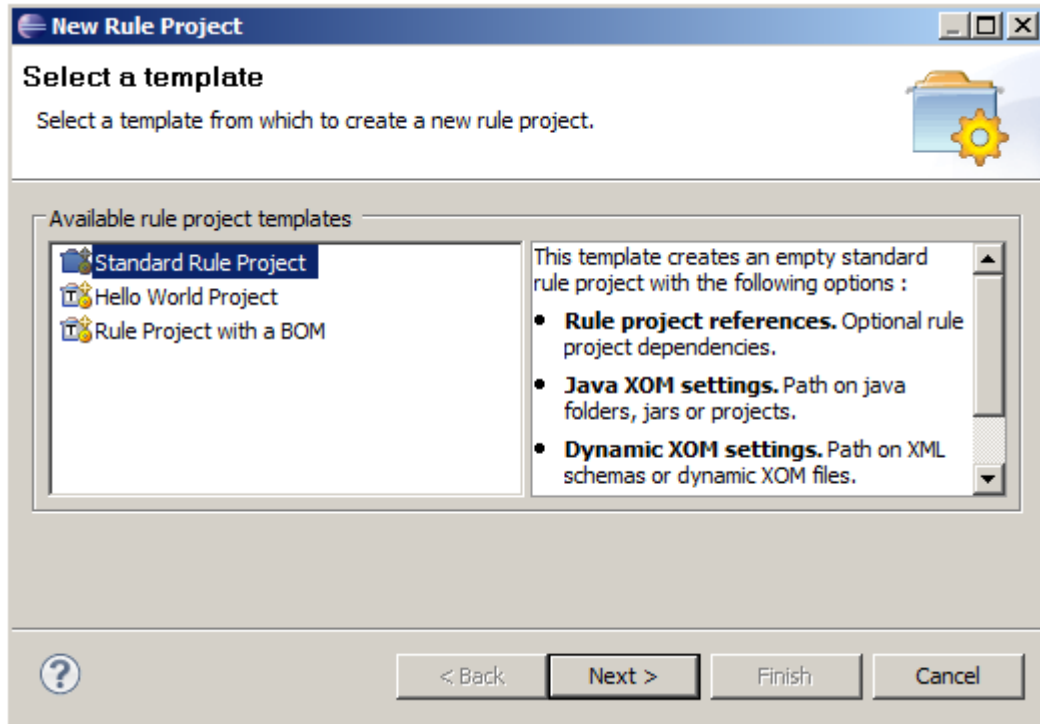
1. Start the IBM Operational Decision Manager Rules Designer from the Start menu. This is an eclipse-based tool, and will start the eclipse Helios workbench. Accept the supplied workspace (c:\workspaces\ODM).

**Important: you must start the Rule Designer from the icon shown on the Windows Start menu. This shortcut has been configured to use the en\_US locale. This matches the configuration of the Decision Center system, which you will use later in the lab.**

2. In the Rule perspective, create a new Rule Project.

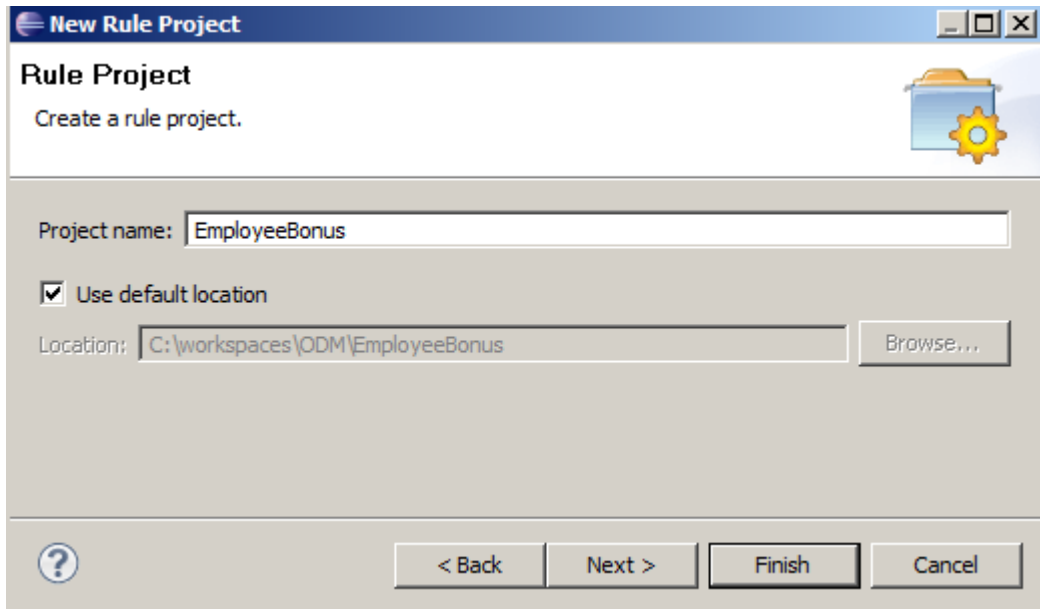


3. Select a Standard Rule Project, and click Next.



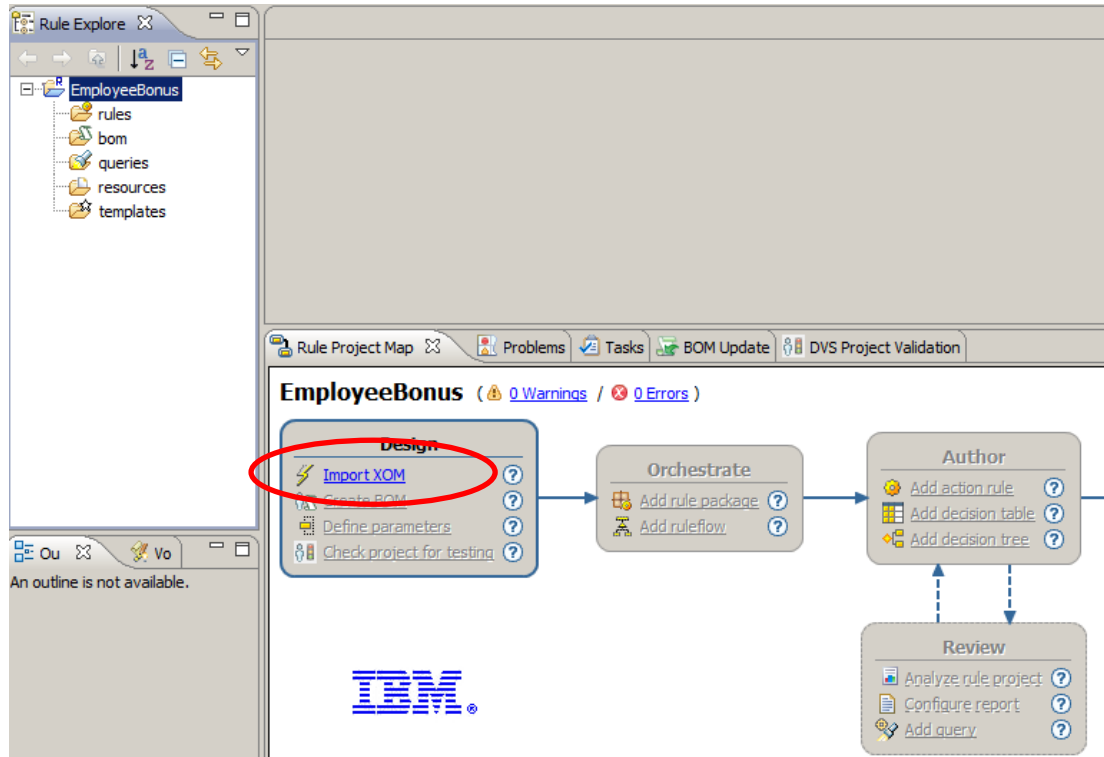
4. Name the project EmployeeBonus, and click Finish.

Note that the project name EmployeeBonus is known as the "ruleset". The Integration Bus application will reference this ruleset, so the precise name and case of this project is important.

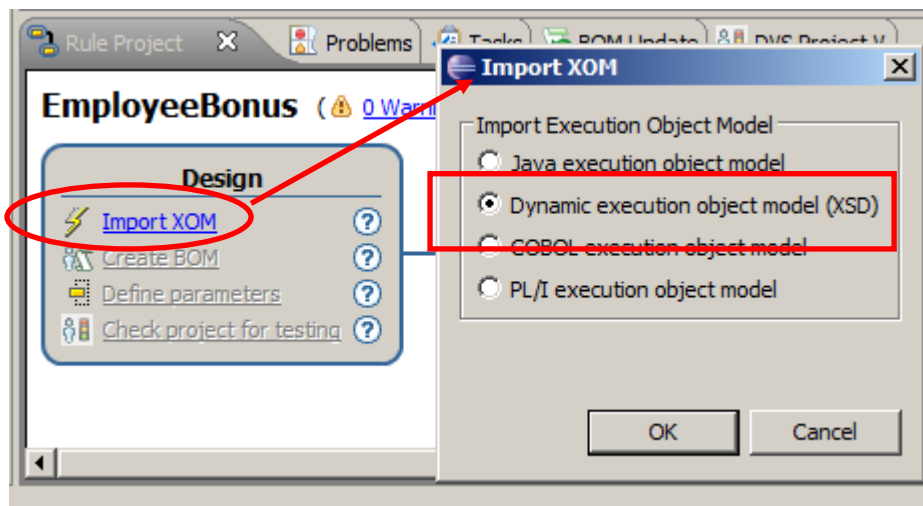


- When the project has been created, the Rule Explorer (navigator) will be populated as normal, and a “wizard-style” set of tasks will be shown in the Rule Project Map view (when the EmployeeBonus project is highlighted in the navigator).

At this stage, the only item available for selection will be “Import XOM”.

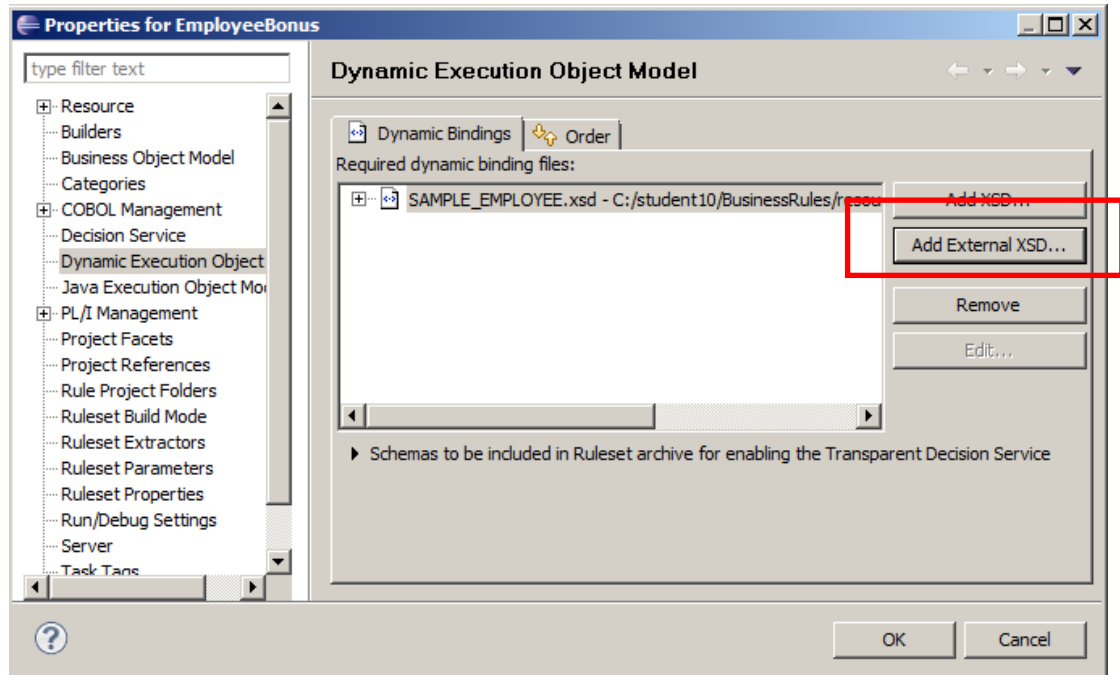


- Click the “Import XOM” link, and then select “Dynamic execution object model (XSD)”. Click OK.



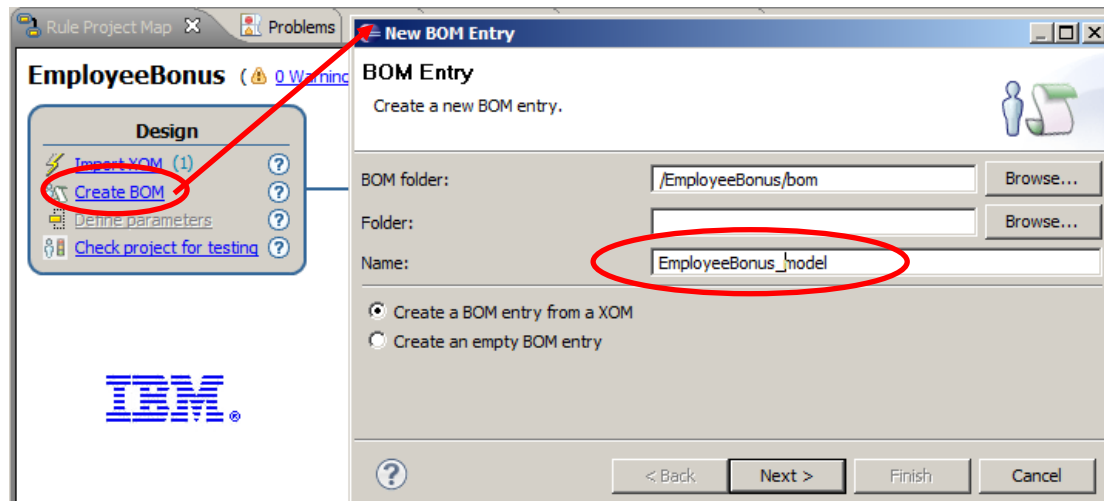
- On the Dynamic Execution Object Model, click “Add External XSD”, and navigate to c:\student10 \ BusinessRules \ resources \ EmployeeServiceInterface \ sample \ iibadmin \ SAMPLE\_EMPLOYEE.xsd.

Click OK.

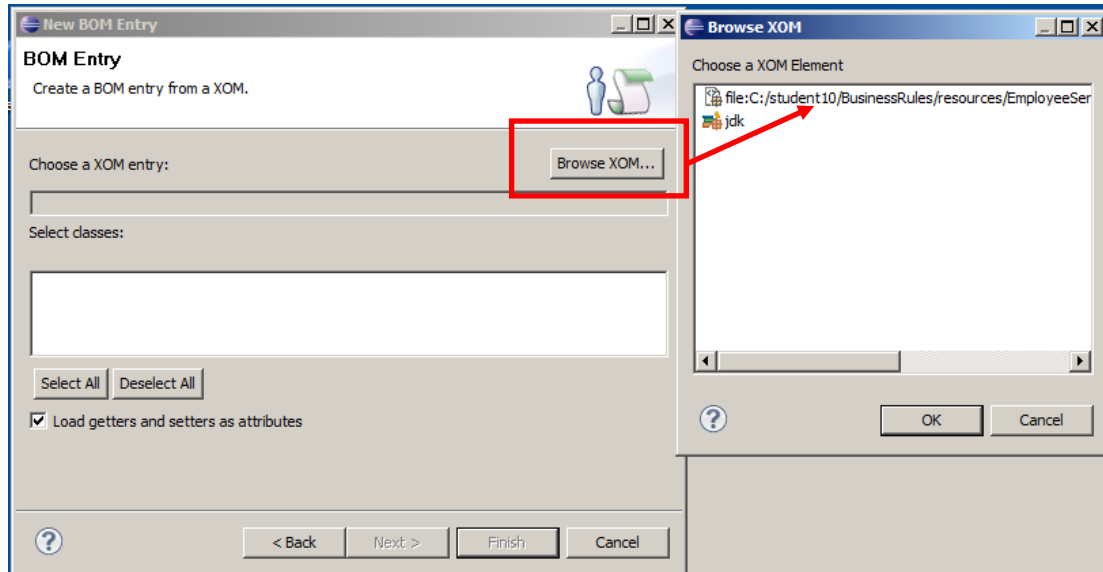


- Back on the Rule Project Map, the “Create BOM” link is now active. Click this, and on the New BOM Entry window, set the name to **EmployeeBonus\_model**.

Click Next.



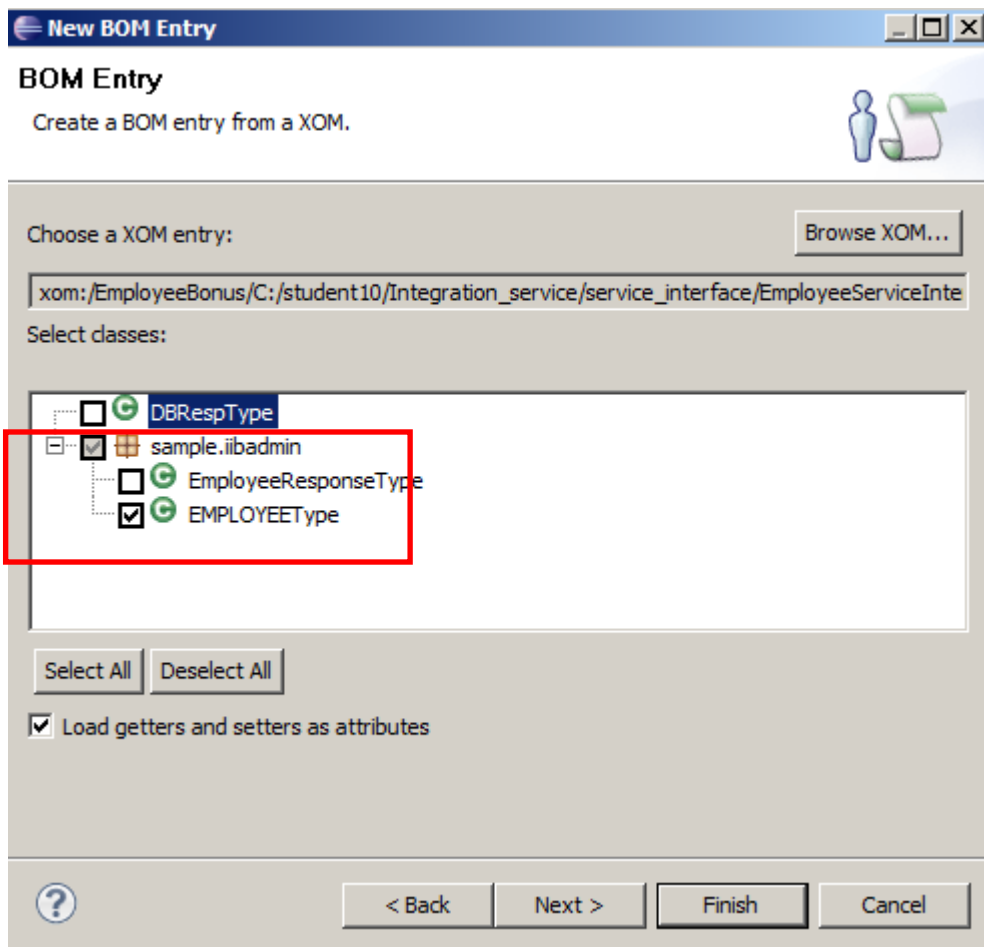
- On the New BOM Entry window, click Browse XOM, and select the SAMPLE\_EMPLOYEE.xsd (XOM) that you just imported. Click OK, then Finish.



- At the BOM Entry window, expand the sample.iibadmin class, and select the EMPLOYEEType.

Click Next.

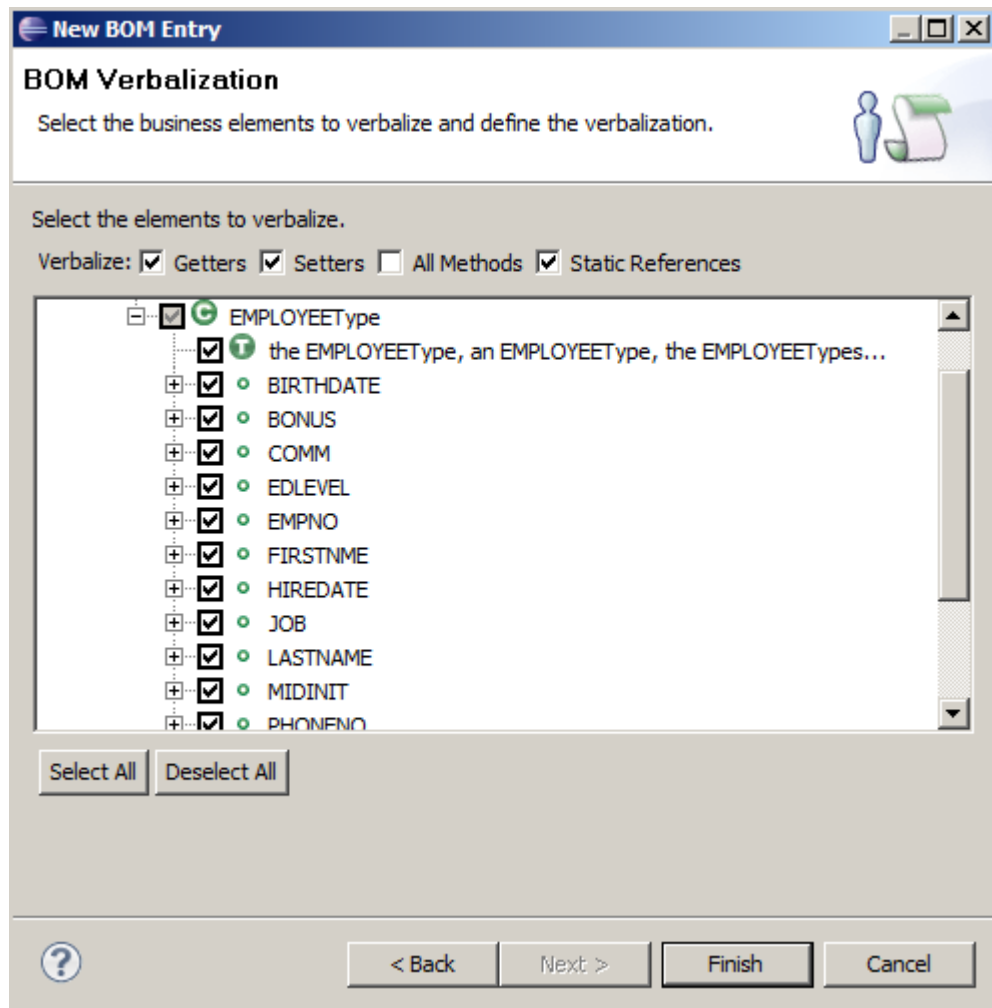
(The DBResp schema was included in the list because it was included in the schema definition in the EmployeeService lab. However, it's not needed to invoke this business rule).



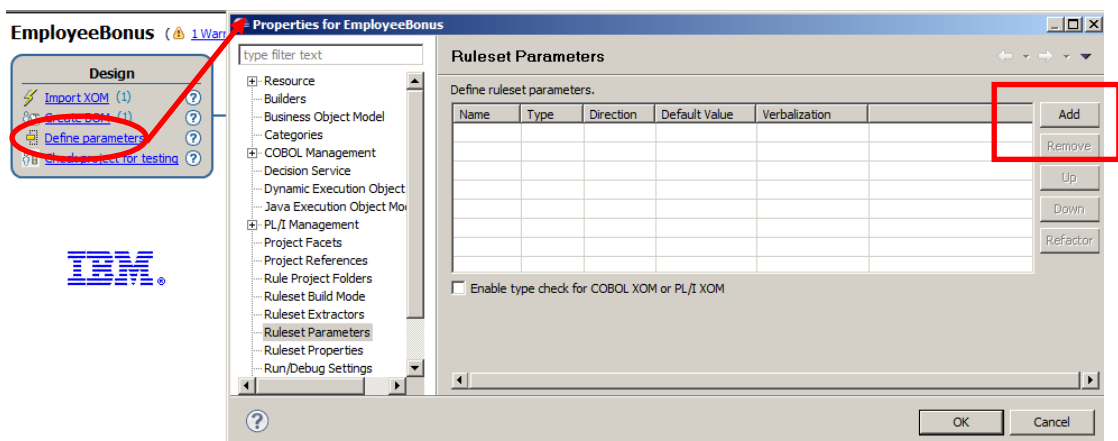


- At the BOM Verbalization window, accept all defaults, and click Finish.

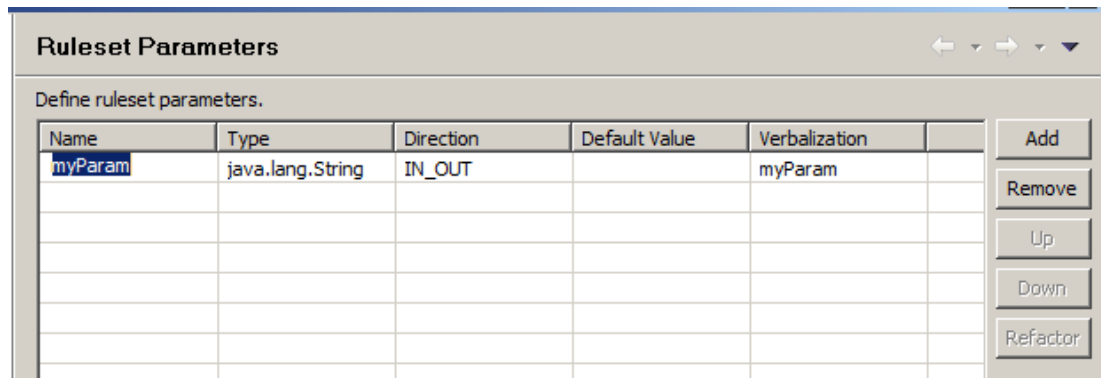
(You could remove some of these elements to make the verbalization simpler; however, we'll leave them in for now).



- Back at the Rule Project Map wizard, the “Define parameters” link will now be active, so create the Ruleset Parameters by clicking Define Parameters. This will open the Properties for EmployeeBonus on the Ruleset Parameters tab.

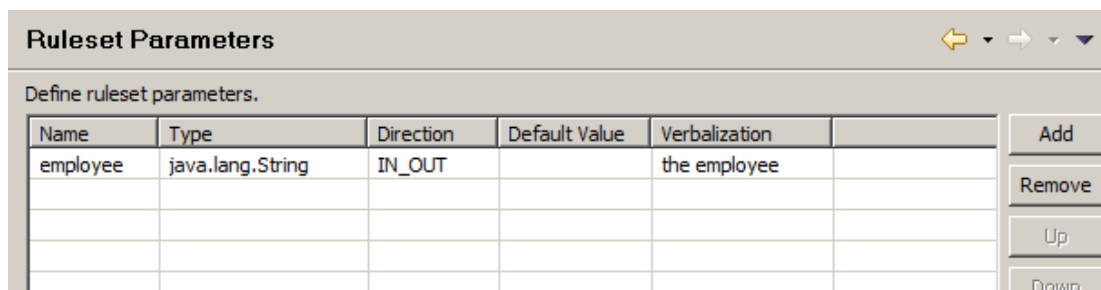


13. Click the Add button. This will add a new parameter with default values.

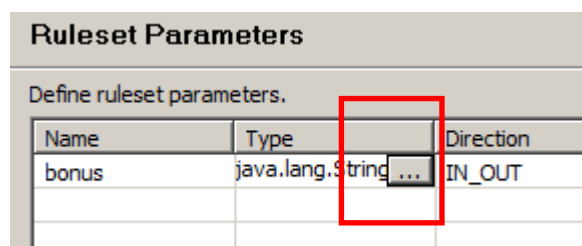


14. The three columns Name, Type and Verbalization need to be changed on this window, as follows:

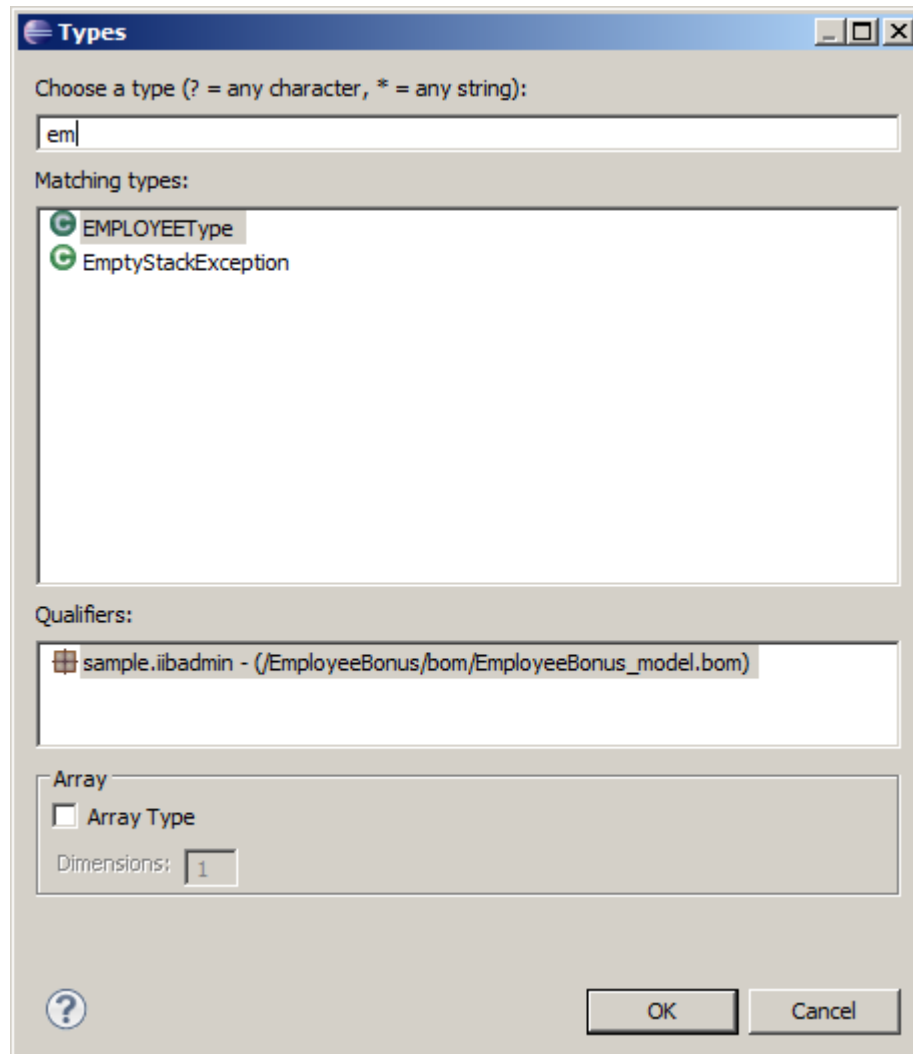
First, change the Name to employee, and change Verbalization to “the employee”.



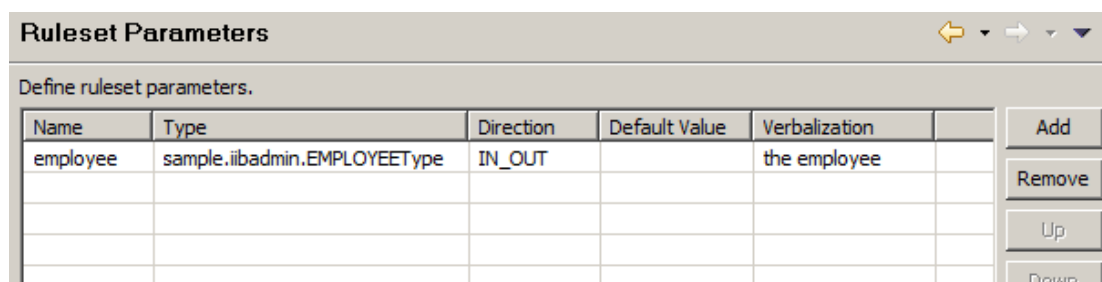
15. Next, click the Type java.lang.String, and click the icon with three dots to invoke the Types Selection editor.



16. On the Types selection editor, type “em” into the filter field, then select EMPLOYEEType, then OK.



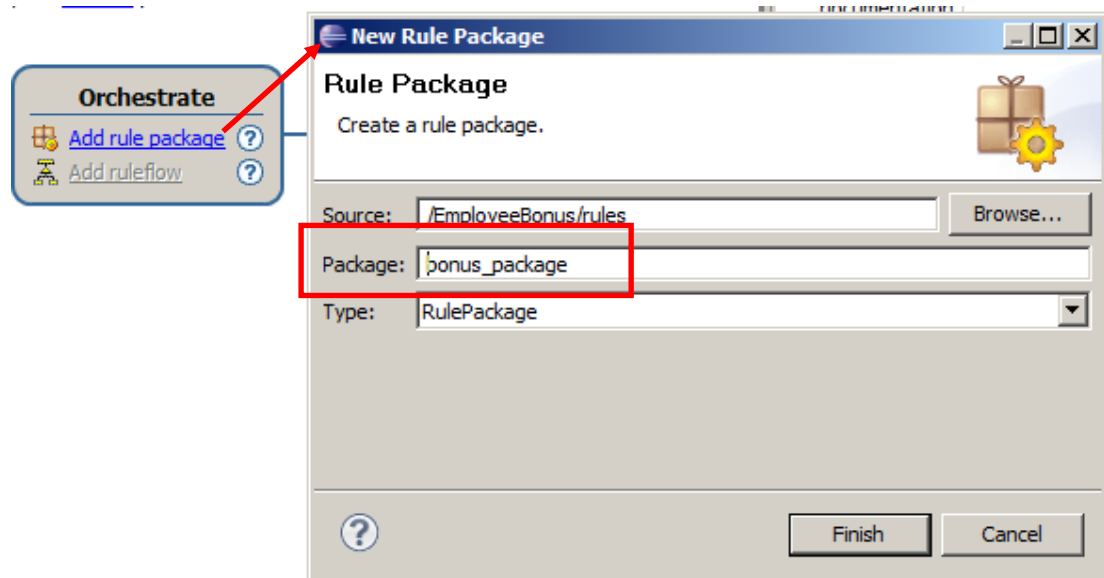
17. When you have finished with the RuleSet parameters, as shown here, click OK.



18. Back at the Rule Project Map, in the Orchestrate section, click "Add rule package".

Name the new package **bonus\_package**.

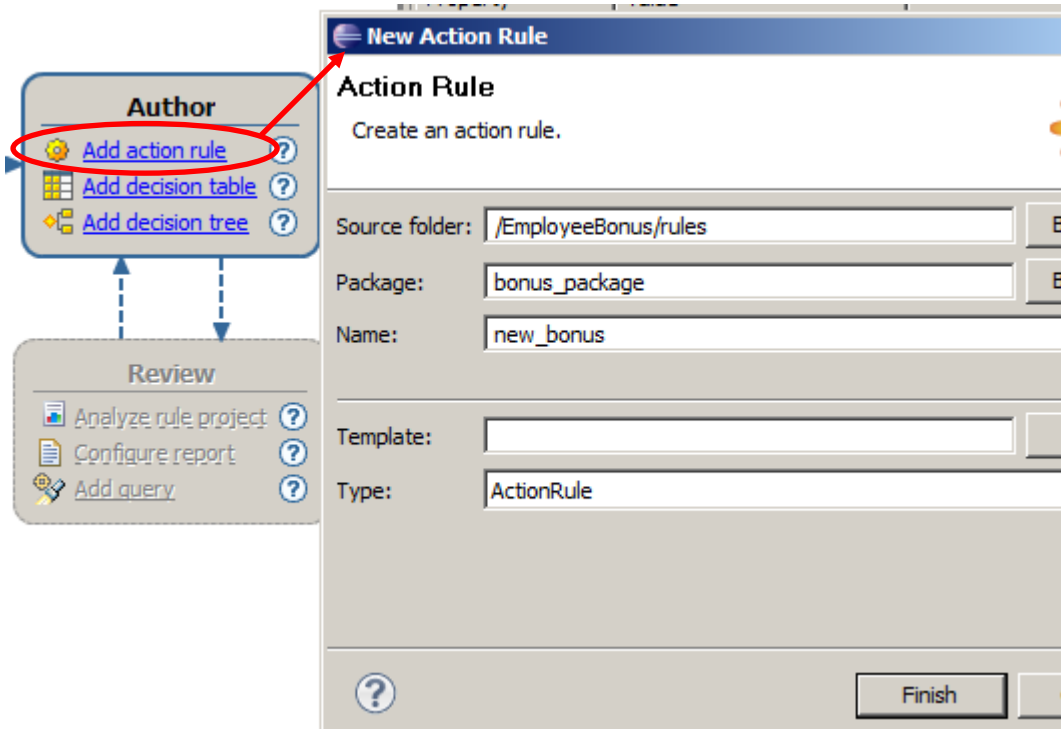
Click Finish.



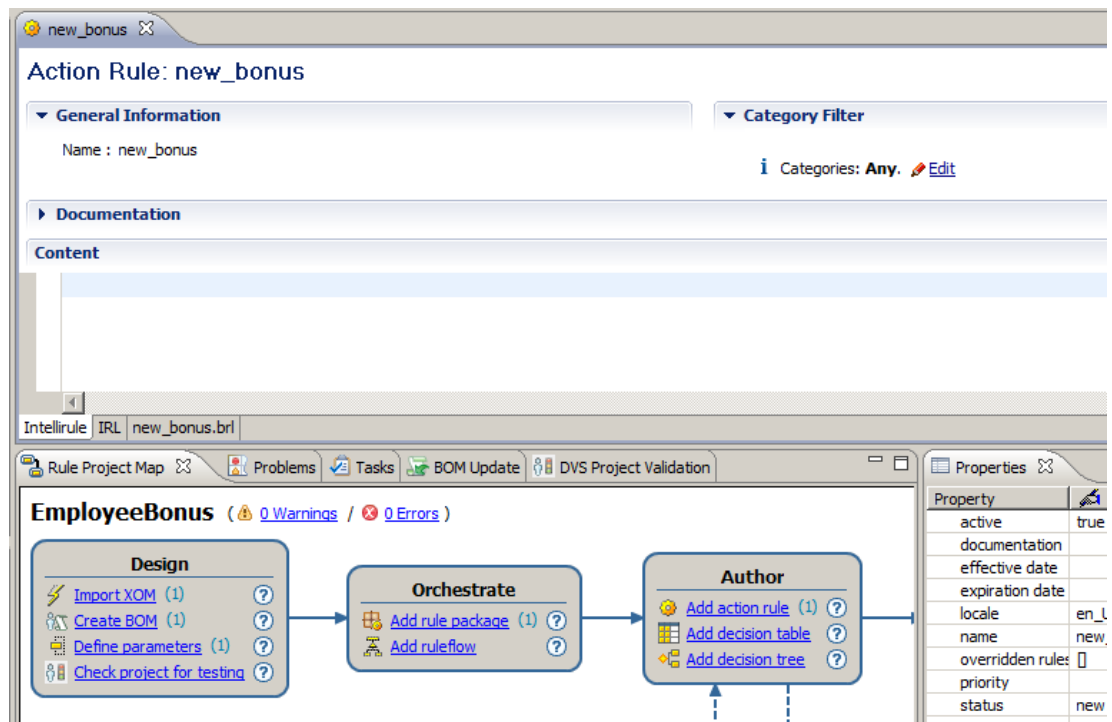
- At the Rule Project Map, in the Author section, click “Add action rule”.

Set the Package to **bonus\_package** (just created), and the name of the new rule to **new\_bonus**.

Click Finish.



- The rule editor will now open with new\_bonus, and you can edit this new rule using the rule editor.

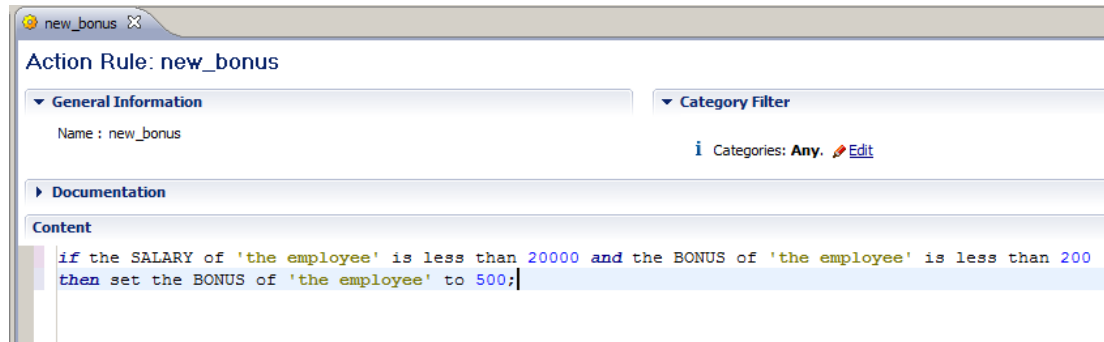


- In the Content pane, using the rule editor and its Content Assist feature, construct the following rule:

```

If the SALARY of 'the employee' is less than 20000 and the BONUS of 'the employee' is less than 200 then set the BONUS of 'the employee' to 500;
    
```

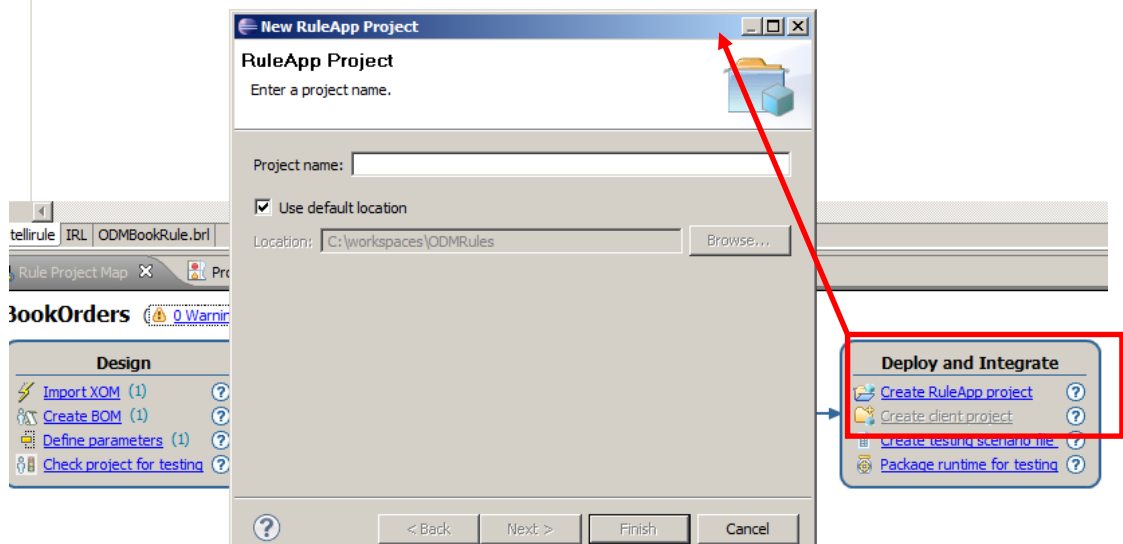
The Content Assist feature uses a combination of the Return (to select a value) and the up/down cursor to move between possible values.



Save the rule.

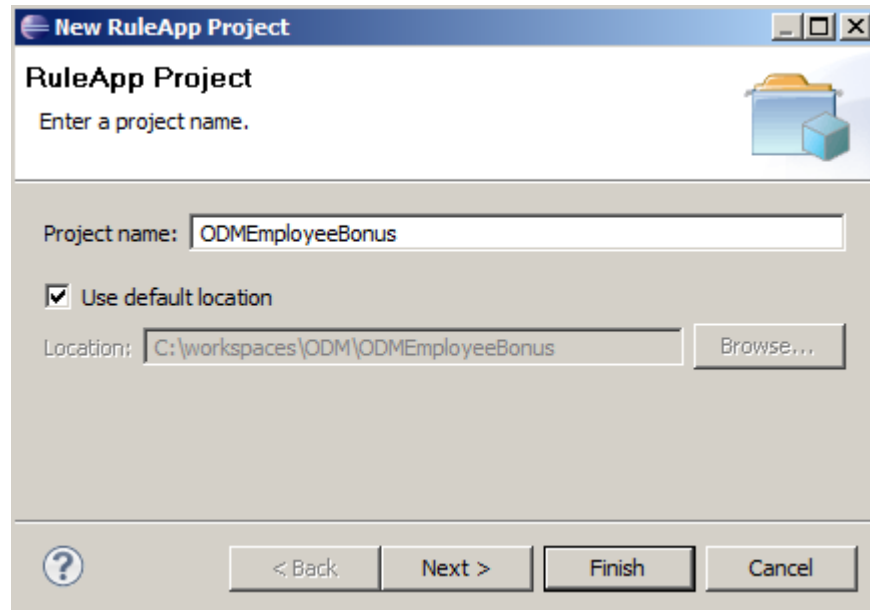
- Finally, you need to create a RuleApp project that can be used to export the rule, so that it can be imported into the Integration Bus Toolkit.

Back at the Rule Project Map, in the Deploy and Integrate section, click “Create RuleApp project”.

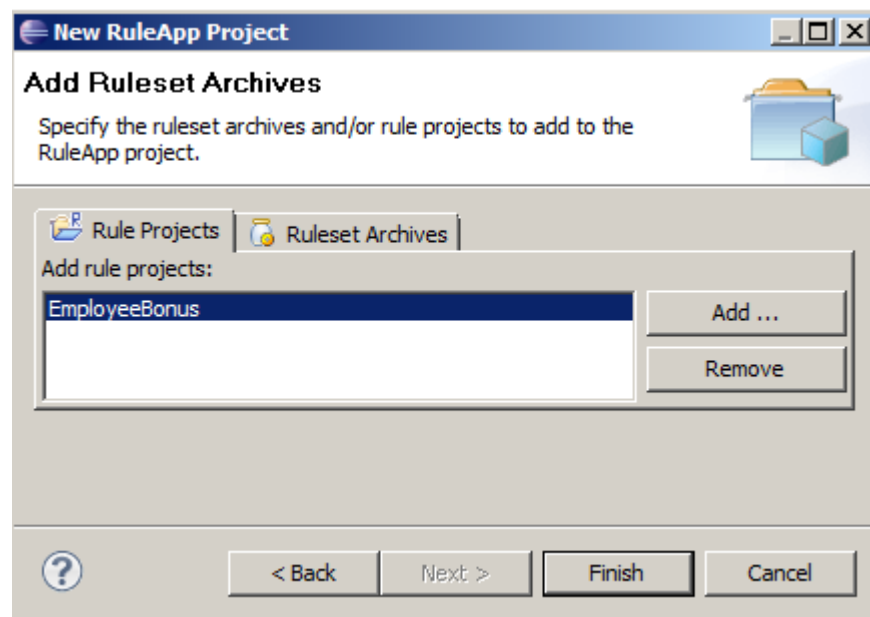


23. Name the project **ODMEmployeeBonus** (case-sensitive); click Next.

Note that this RuleApp is the ODM container that holds the ruleset, and is the way that the Integration Bus application will reference the required ruleset. Later in this lab, you will create the corresponding deployment container for the RuleApp into the ODM runtime, which will also named ODMEmployeeBonus, enabling the ODM system to locate the requested ruleset.



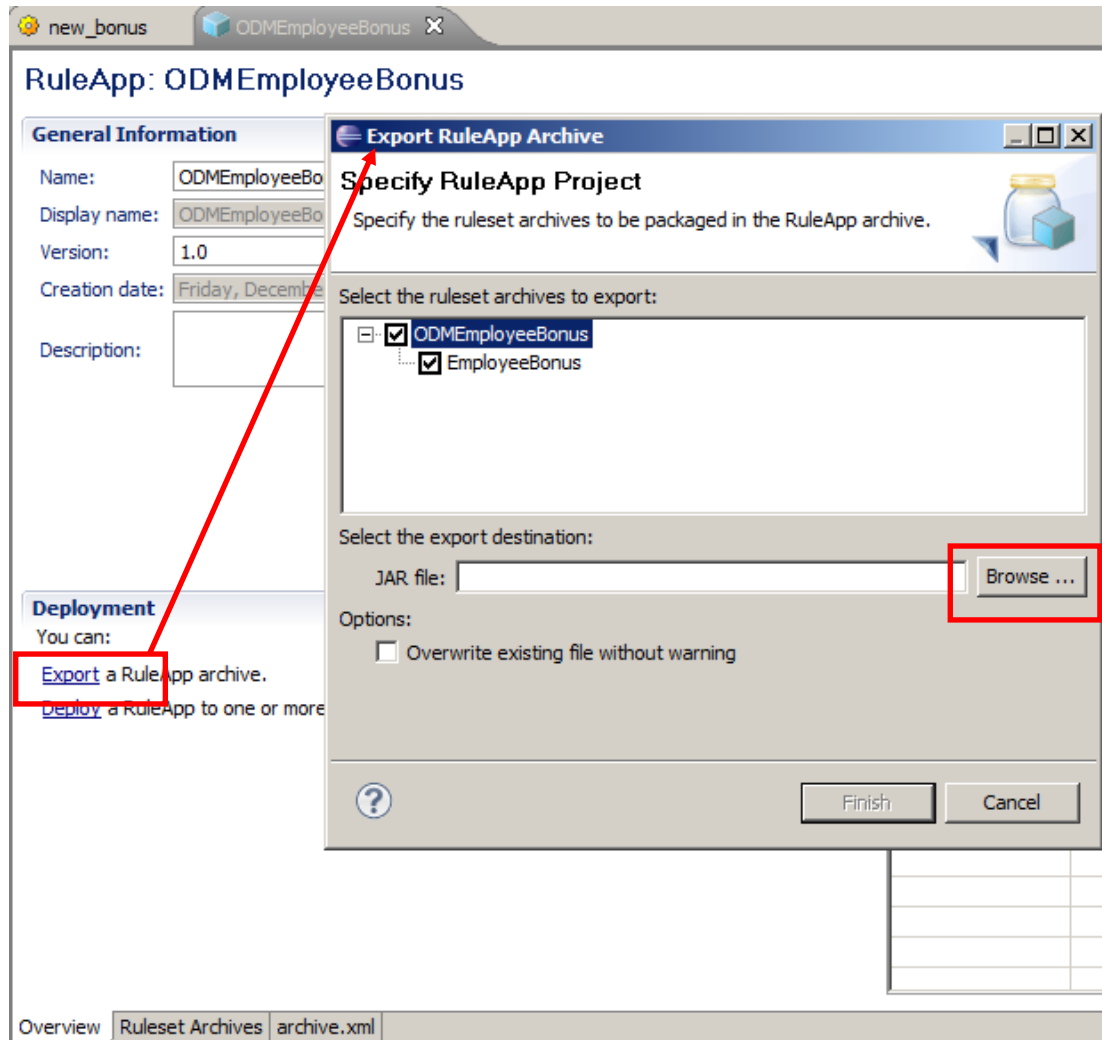
24. Ensure the Rule Projects tab is active, and select EmployeeBonus. Click Finish.



## 2.2 Export the RuleApp Archive

In this section, you will export the RuleApp so that it can be used by the Integration Bus EmployeeService. It is exported in the form of a jarfile.

1. On the ODMEmployeeBonus project, click “Export” a RuleApp archive.

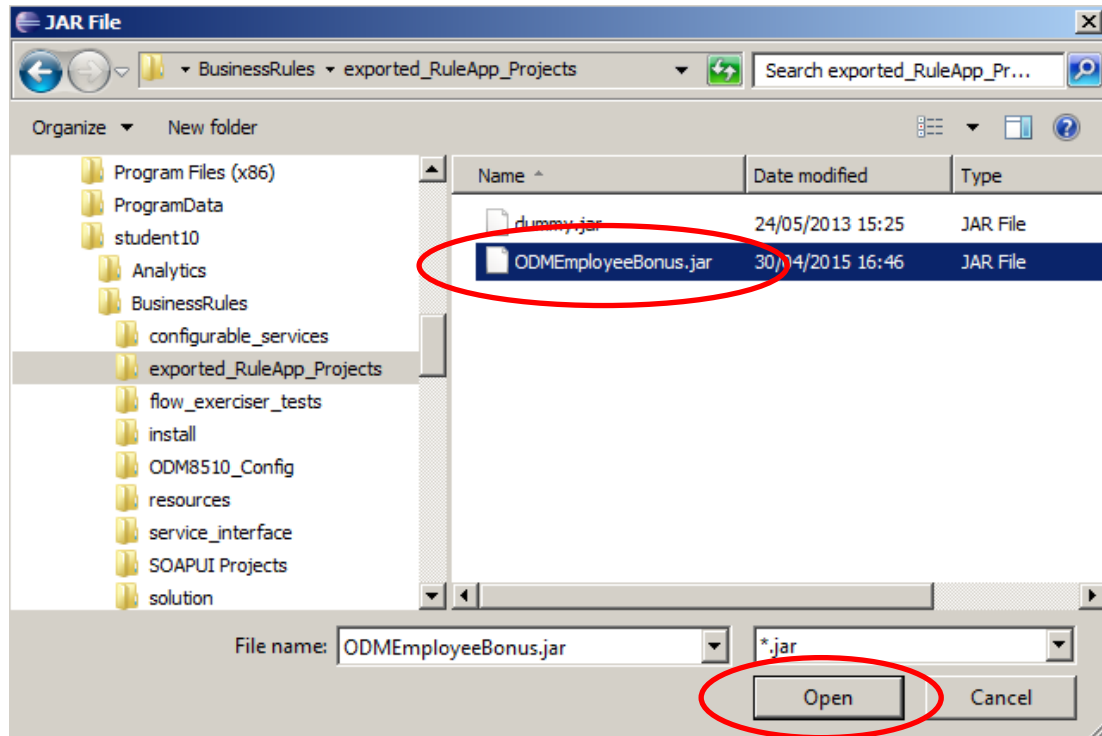




- The file has to be exported as a jar file, so use the Browse button to navigate to

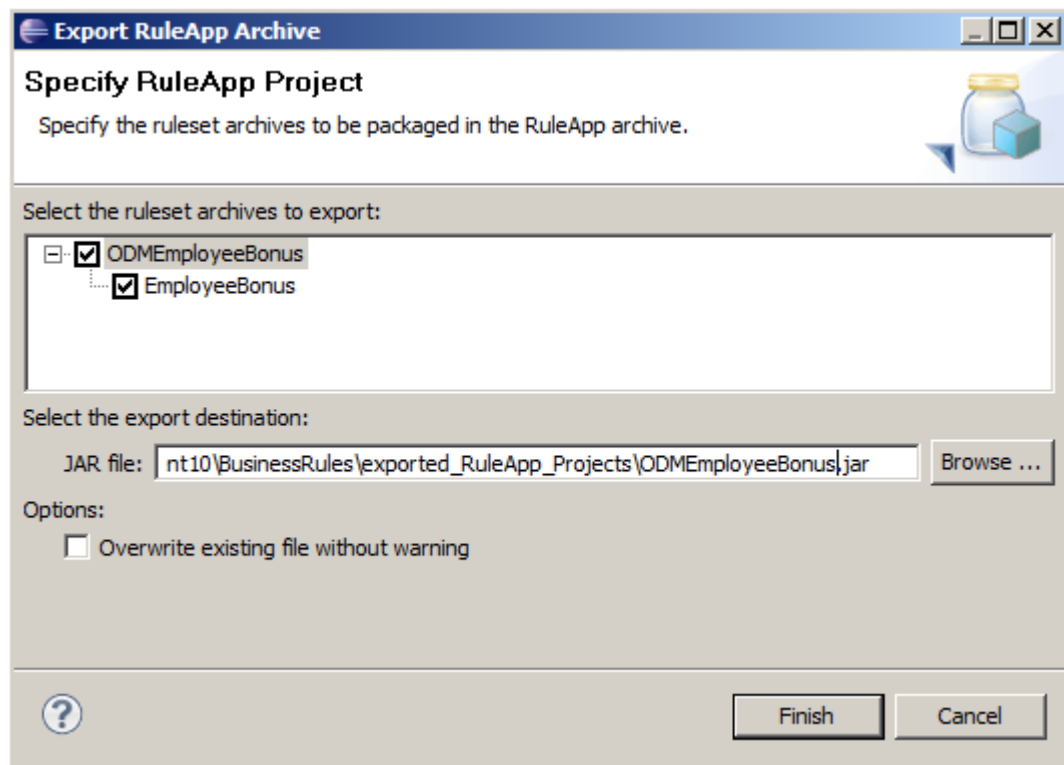
```
c:\student10 \ BusinessRules \ exported_RuleApp_Projects
```

You will see that there is already a file called ODMEmployeeBonus.jar in this folder. Highlight this, and then click Open (you are going to overwrite this file with your new version).



- The fully-qualified file name will now be set correctly in the Export Archive window, so click Finish.

Answer YES when asked to overwrite the existing file.



- Summary so far:

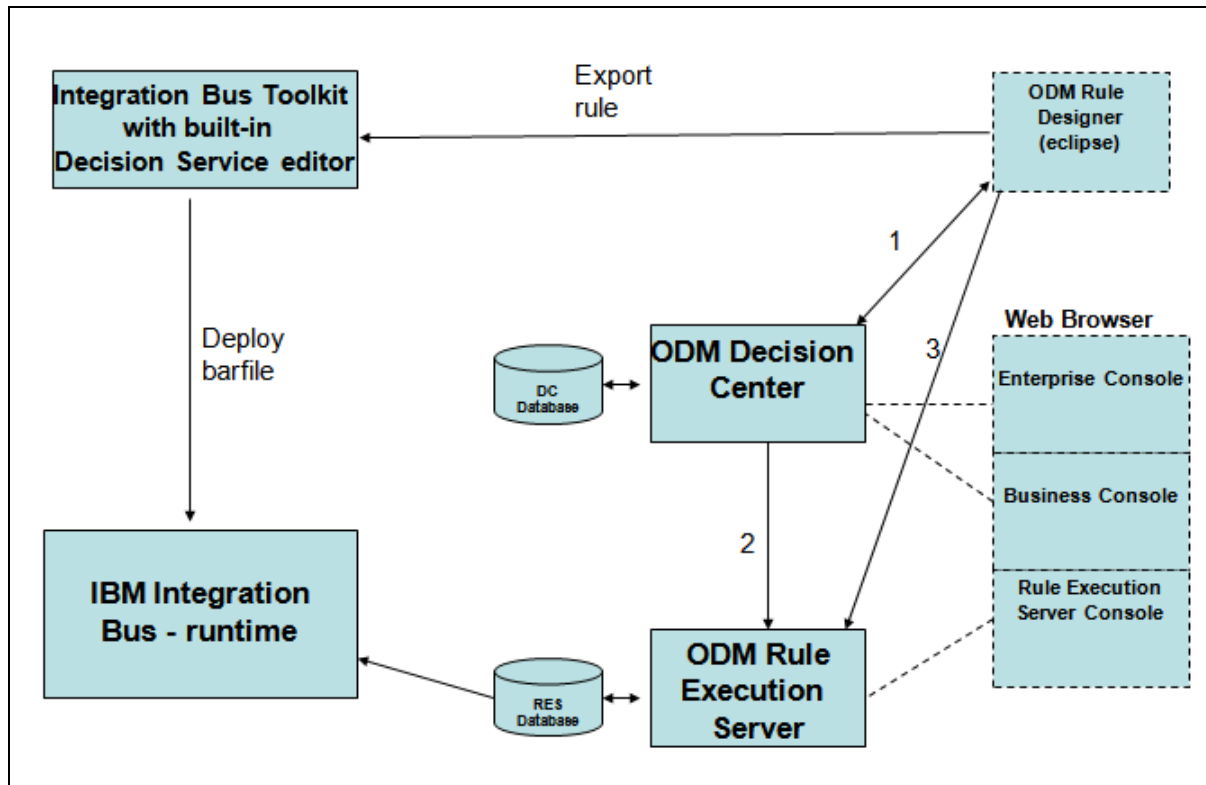
This has completed the definition of the rule and ruleset in the ODM Rule Designer.

You have also exported the ruleset, in the form of a RuleApp named ODMEmployeeBonus, and stored this as a jar file on the local filesystem. This ruleset will be imported into the IIB Toolkit in chapter 3.

You will now go on to store this ODM project in the ODM Decision Center. From here, it will be deployed into the Rule Execution Server.

## 2.3 Add the ODM Rule to the ODM Decision Center Repository

As a reminder, at runtime the ODM rules and rulesets are stored and loaded by the Integration Bus from the Rule Execution Server database, as shown in the following schematic.



When using ODM rules, the rule is developed initially in the eclipse Rule Designer. You can either load this rule into the Decision Center database (step 1), and then deploy the rule (contained within a RuleApp) from there into the Rule Execution Server (step 2). Or you can deploy the rule direct from the Rule Designer into the Rule Execution Server (step 3).

This lab will use steps 1 and 2. You can update the rule using the Enterprise Console web browser, and then deploy the rule using the same console, although this lab will not include that scenario.

In this lab, you will update the rule with the ODM Business Console, and deploy the rule using the web browser Enterprise Console.

## 2.3.1 Step 1 – Add the rule into the Decision Center repository

The rule that you have created in the Rule Designer now needs to be stored in the ODM repository, which is owned by the Decision Center component of IBM Operational Decision Management. ODM runs as a WebSphere Application Server application, so WAS must be started on your system. (For WAS gurus, we have created a WAS profile named ODMSamples8510).

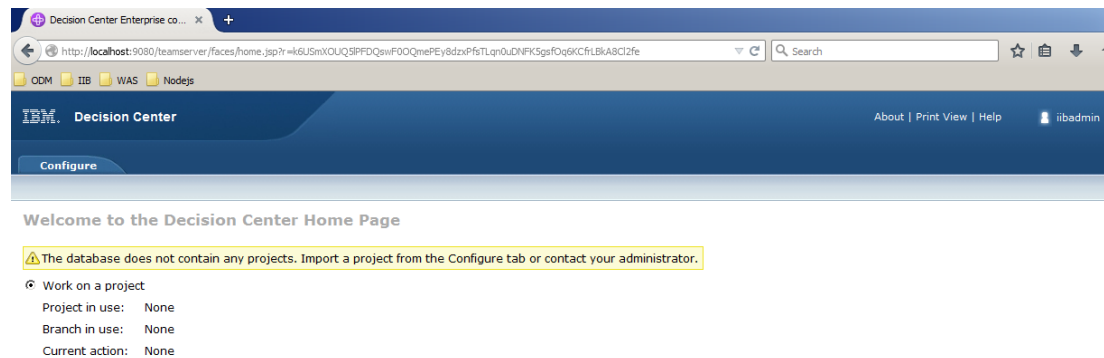
**If you are running this lab in a hosted workshop, we have already started WAS for you. If you are running it independently, start WAS by using the desktop icon.**

1. First, login to the Decision Center Enterprise Console to see what projects are currently defined.

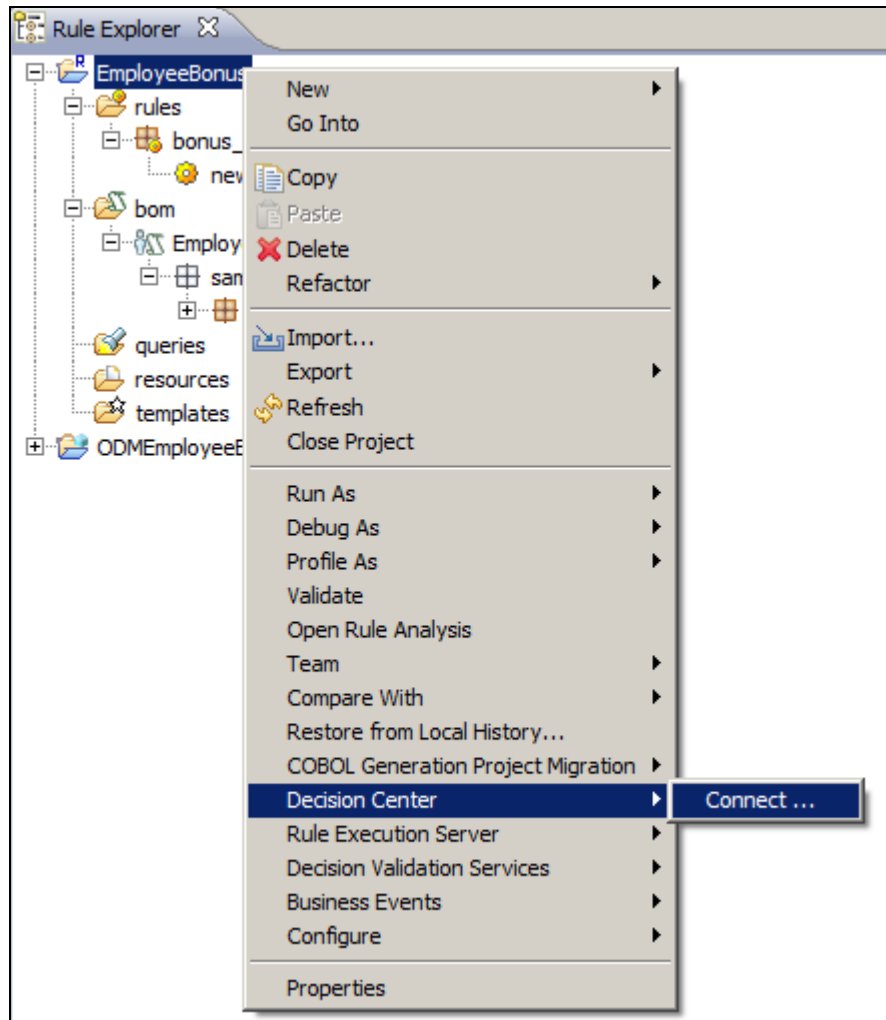
In the Firefox browser, use the Decision Center Enterprise Console (bookmark in the ODM bookmark folder).

Login with the user iibadmin/passw0rd. You will see that there are no projects currently defined in the ODM database.

Logout from the Enterprise Console.



2. In the eclipse Rule Designer, right-click the EmployeeBonus project and select Decision Center, Connect.



- Set the following connection properties (they may already be set by default as a result of our earlier testing).

**NB: all these values are case-sensitive.**

URL: <http://localhost:9080/teamserver>  
User name: iibadmin  
Password: passw0rd  
Data source: jdbc/iIogDataSource

Click Connect. Assuming the connection is successful, click Finish.

**Decision Center configuration**  
Configure the project to be synchronized with a Decision Center project.

Connection

URL:

User name:

Password:

Data source:

Synchronization Filter

Filter rule elements to be synchronized using the specified query

Project configuration

Create a new project on Decision Center

Synchronize with the existing Decision Center project

- When you are successfully connected, you will see the following (small) message:

**Decision Center configuration**

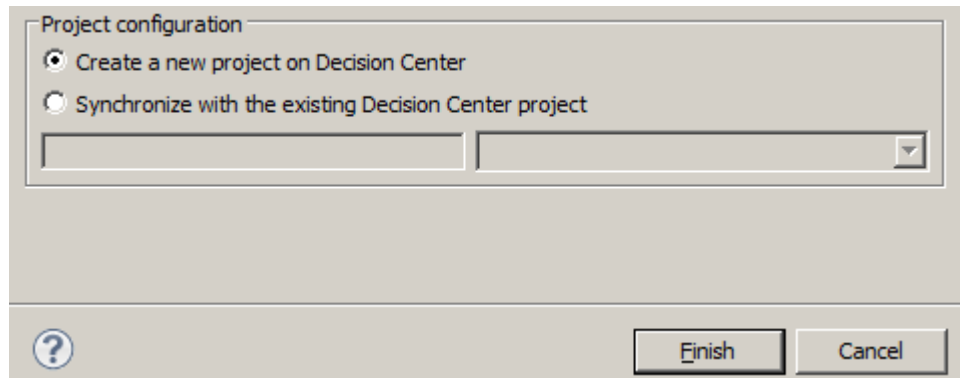
Connection with Decision Center established.

Connection

URL:

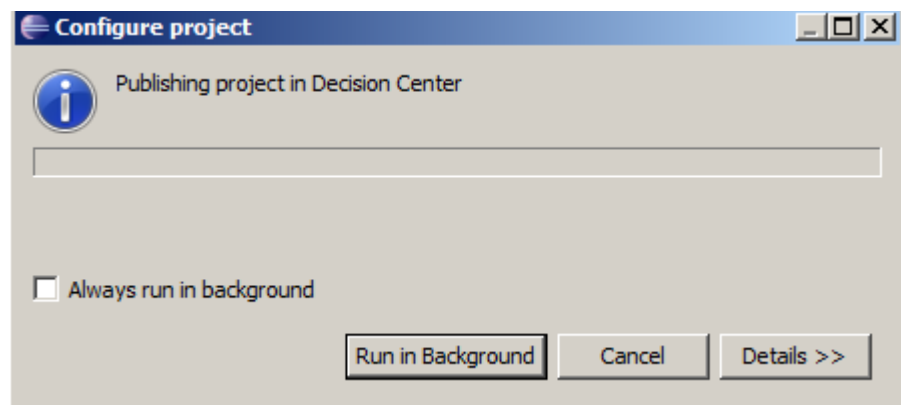
User name:

5. You may see this window. If you do, ensure you create a new project (the default), and click Finish.

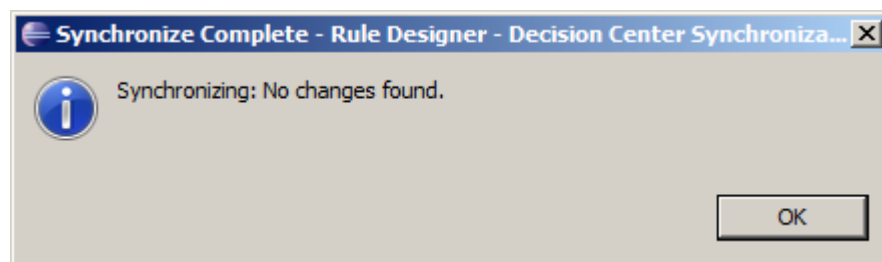


The Rules Designer will then connect to the Decision Center and synchronize the rule.

A number of progress windows will appear, similar to this:



If this is successful, the final window should be as shown. Note that even though the message says that no changes have been found, the project should have been published to the Decision Center, so click OK.



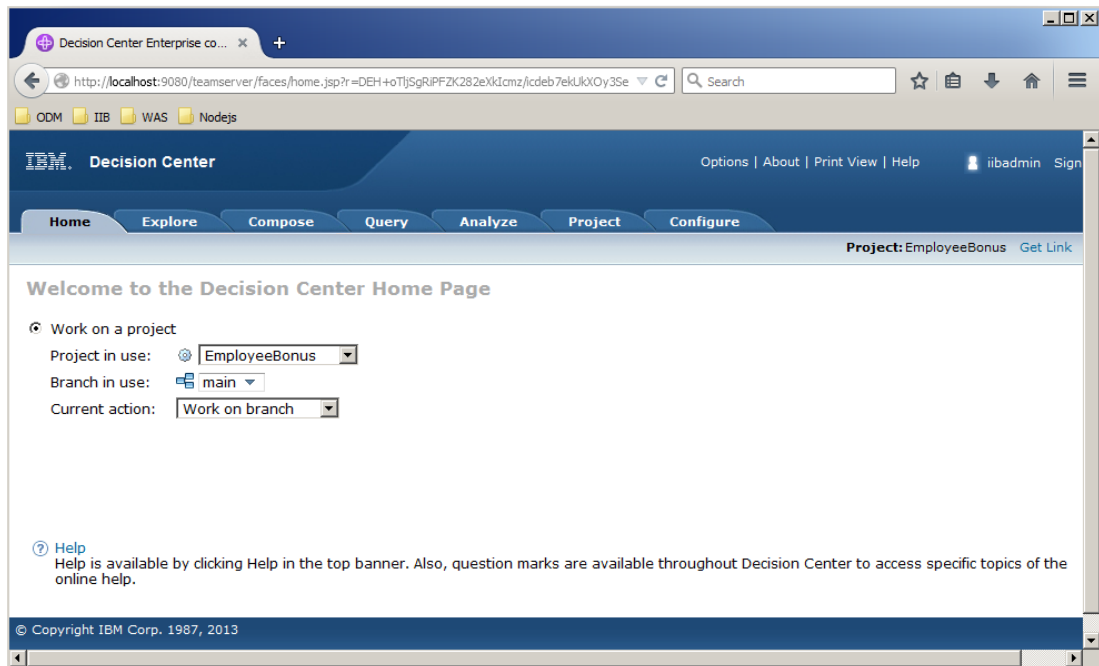
You may see a warning about opening the Team Sync perspective. Answer Yes to this.

## 2.3.2 Step 2 – Deploy the rule from the Decision Center to the Rule Execution Server

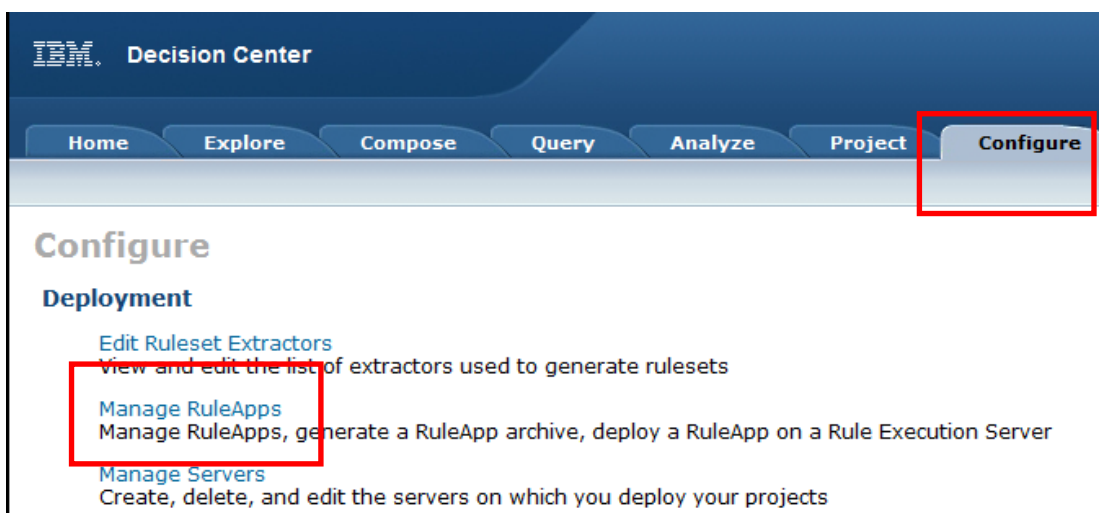
1. Login to the Decision Center Enterprise Console again (iibadmin/passw0rd).

You will now see that the EmployeeBonus project has been checked into the ODM database.

As a general reference, the Decision Center Console usually places the name of the active project in the top right of the browser window (in this case EmployeeBonus).

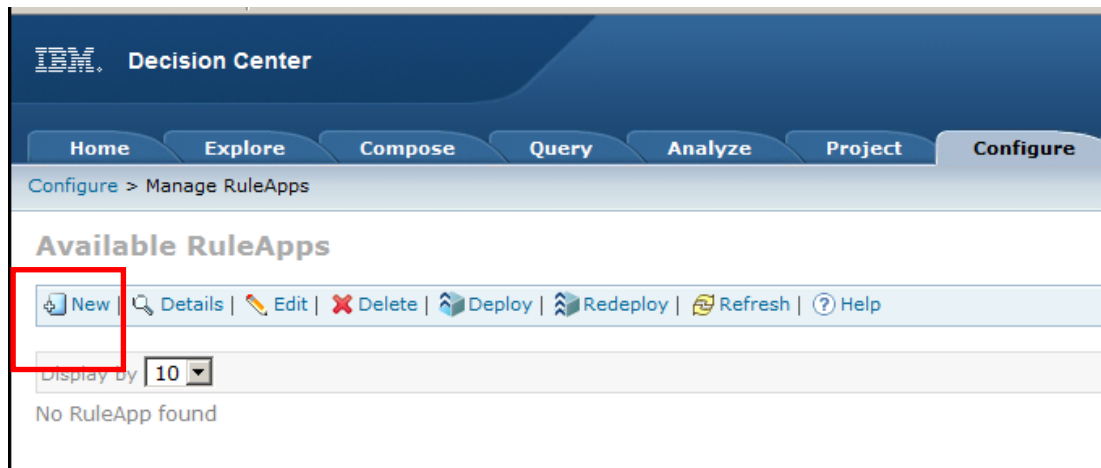


2. To deploy the rule to the Rule Execution Server, you first have to create a RuleApp to contain the rule.  
Click the Configure tab, and then click “Manage RuleApps”.



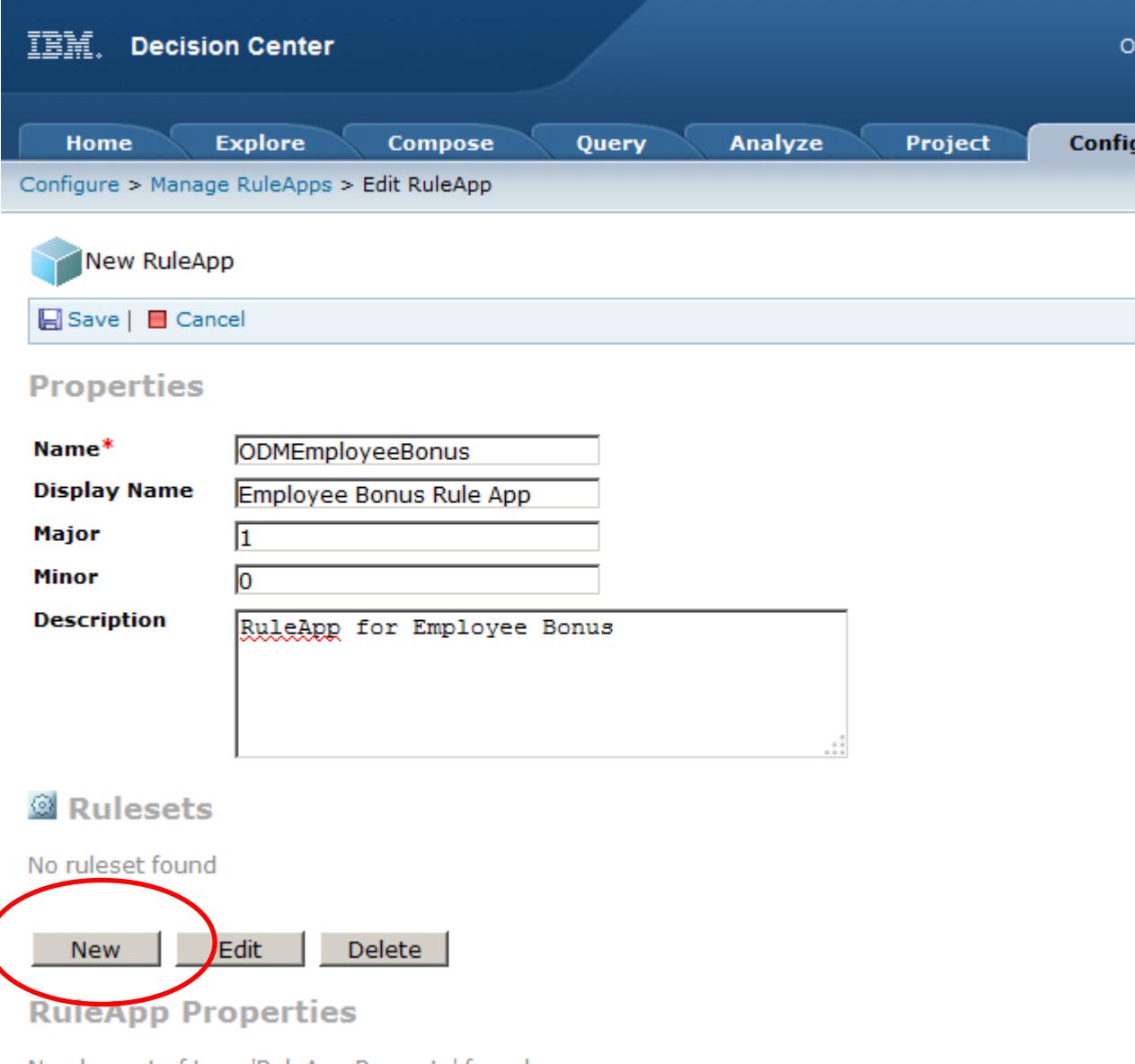


3. No RuleApps are defined at the moment, so click New to create a new one.



4. Set the following values:
  - **Name:** **ODMEmployeeBonus** (This has to match the RuleApp Project we created earlier in the Rule designer, so precise text and case is important)
  - **Display Name:** **Employee Bonus Rule App**
  - **Major:** **1**

The new RuleApp needs to contain a Ruleset. However, we do not have any Rulesets at this point, so you need to create one. Click New under the Rulesets paragraph.



IBM Decision Center

Home Explore Compose Query Analyze Project Config

Configure > Manage RuleApps > Edit RuleApp

New RuleApp

Save | Cancel

### Properties

<b>Name*</b>	ODMEmployeeBonus
<b>Display Name</b>	Employee Bonus Rule App
<b>Major</b>	1
<b>Minor</b>	0
<b>Description</b>	RuleApp for Employee Bonus

### Rulesets

No ruleset found

New Edit Delete

### RuleApp Properties

5. Specify the following values for the Ruleset:
- Name: **EmployeeBonus** (name and case are important)
  - Display name: Employee bonus ruleset
  - Project: EmployeeBonus
  - Major: 1

Click the Save link.

The screenshot shows the IBM Decision Center interface. The breadcrumb trail is 'Configure > Manage RuleApps > Edit RuleApp > Edit Ruleset'. The page title is 'EmployeeBonus'. A red circle highlights the 'Save' and 'Cancel' buttons. Below this is the 'Properties' section with the following fields:

Name*	EmployeeBonus
Display Name	Employee bonus ruleset
Description	
Project*	EmployeeBonus
Baseline or Branch*	main
Extractor	<none>
Major	1
Minor	0
Enabled	<input checked="" type="checkbox"/>
Debug	<input type="checkbox"/>

Below the properties is the 'Ruleset Properties' section, which contains the message: 'No element of type 'Ruleset Property' found'. At the bottom are three buttons: 'New', 'Edit', and 'Delete'.

- 6. This will return you to the New RuleApp dialogue. Click the Save link.

**IBM Decision Center**

Home Explore Compose Query Analyze Project Config

Configure > Manage RuleApps > Edit RuleApp

New RuleApp

Save | Cancel

⚠ Changes have been made to the RuleApp. Click 'Save' to save your changes

**Properties**

**Name\*** ODMEmployeeBonus

**Display Name** Employee Bonus Rule App

**Major** 1

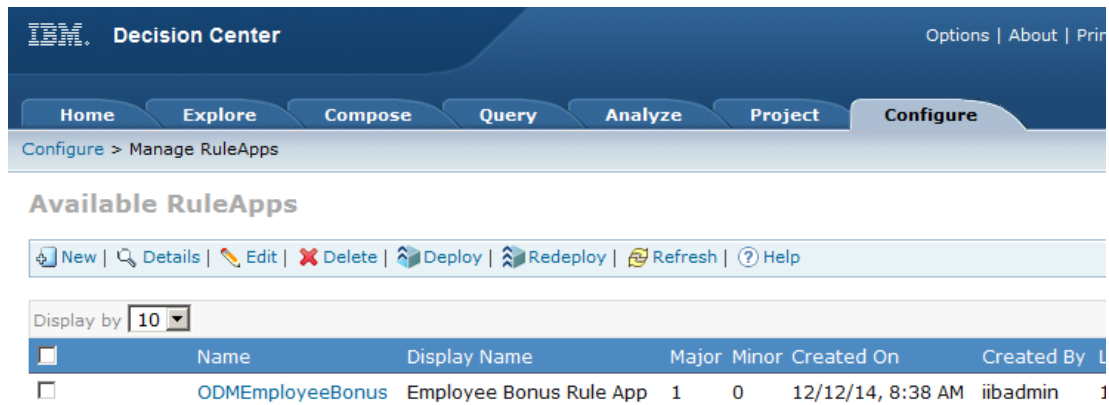
**Minor** 0

**Description** RuleApp for Employee Bonus

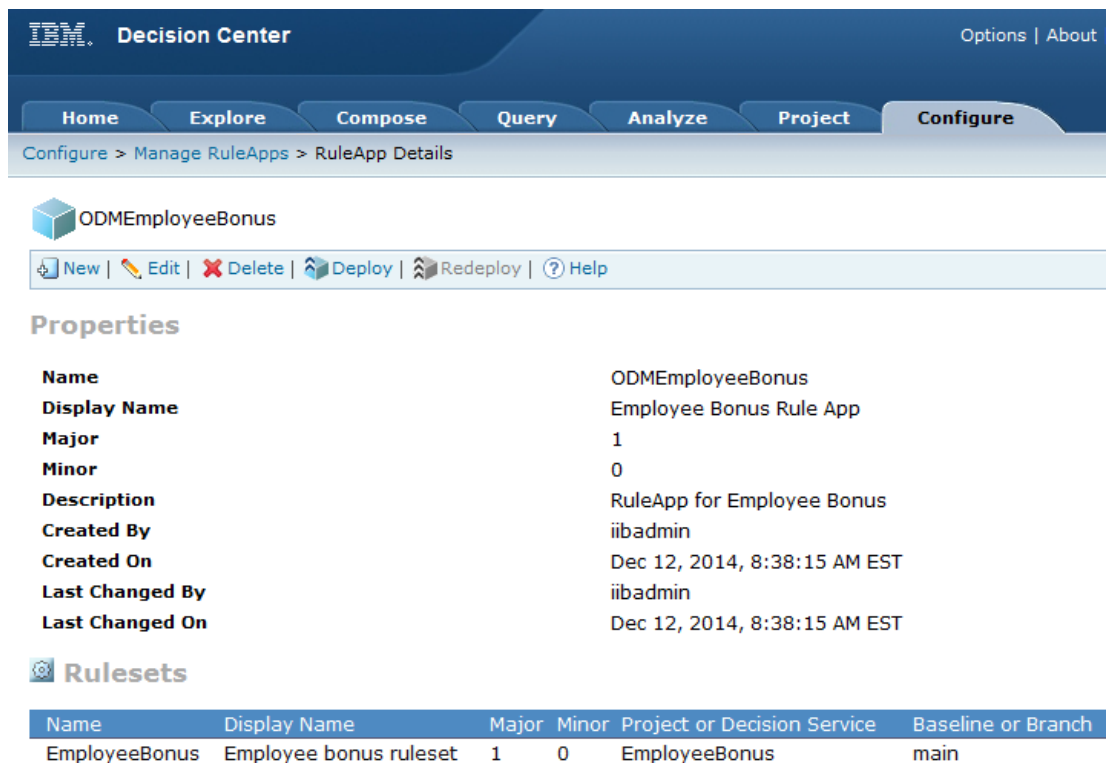
**Rulesets**

<input type="checkbox"/>	Name	Display Name	Major	Minor	Project or Decision Service	Base
<input type="checkbox"/>	EmployeeBonus	Employee bonus ruleset	1	0	EmployeeBonus	main

- The new RuleApp ODMEmployeeBonus will have been defined.

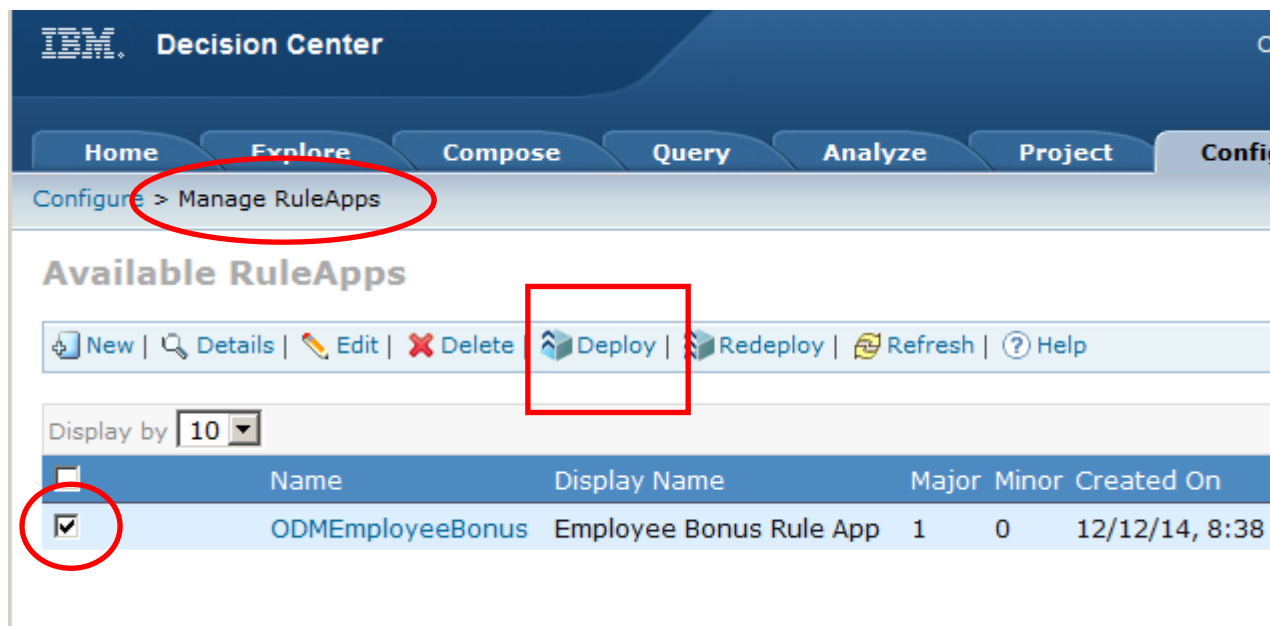


- If you want, you can click ODMEmployeeBonus, and this will show you the Ruleset that it contains.



- Return to the RuleApp display (click "Manage RuleApps" on the breadcrumb trail if you have lost position).

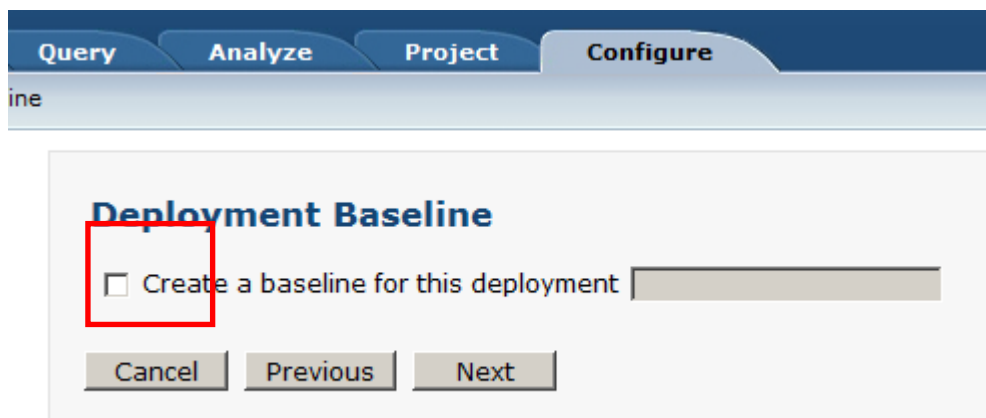
Select ODMEmployeeBonus and click Deploy.



The screenshot shows the IBM Decision Center interface. The breadcrumb trail is "Configure > Manage RuleApps", with "Manage RuleApps" circled in red. Below the breadcrumb is a toolbar with buttons for "New", "Details", "Edit", "Delete", "Deploy", "Redeploy", "Refresh", and "Help". The "Deploy" button is circled in red. Below the toolbar is a "Display by" dropdown menu set to "10". A table lists available RuleApps. The first row is selected, and its checkbox is circled in red.

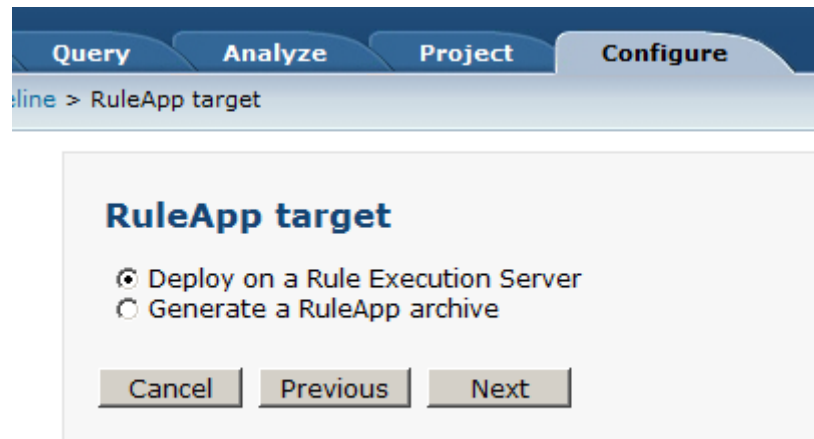
<input type="checkbox"/>	Name	Display Name	Major	Minor	Created On
<input checked="" type="checkbox"/>	ODMEmployeeBonus	Employee Bonus Rule App	1	0	12/12/14, 8:38

- Deselect the baseline selection, click Next.



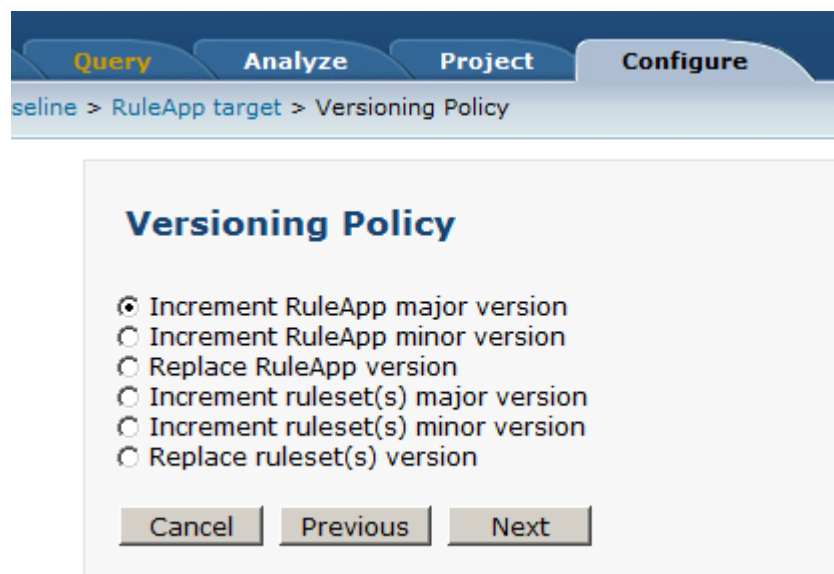
The screenshot shows the "Deployment Baseline" dialog box. The "Configure" tab is selected. The dialog contains a checkbox labeled "Create a baseline for this deployment" which is circled in red. Below the checkbox are three buttons: "Cancel", "Previous", and "Next".

11. Select "Deploy on a Rule Execution Server", click Next.



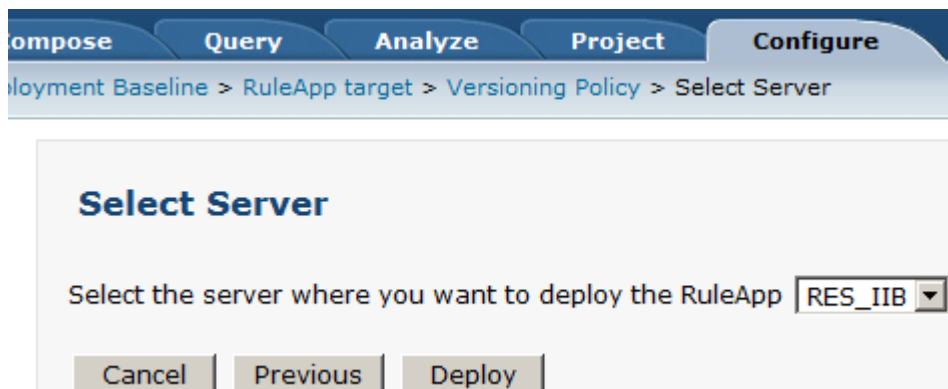
The screenshot shows a configuration dialog titled "RuleApp target". At the top, there are four tabs: "Query", "Analyze", "Project", and "Configure", with "Configure" being the active tab. Below the tabs, the breadcrumb path is "Deployment Baseline > RuleApp target". The main content area has the heading "RuleApp target" and two radio button options: "Deploy on a Rule Execution Server" (which is selected) and "Generate a RuleApp archive". At the bottom, there are three buttons: "Cancel", "Previous", and "Next".

12. Select "Increment RuleApp major version", click Next.



The screenshot shows a configuration dialog titled "Versioning Policy". At the top, there are four tabs: "Query", "Analyze", "Project", and "Configure", with "Configure" being the active tab. Below the tabs, the breadcrumb path is "Deployment Baseline > RuleApp target > Versioning Policy". The main content area has the heading "Versioning Policy" and six radio button options: "Increment RuleApp major version" (selected), "Increment RuleApp minor version", "Replace RuleApp version", "Increment ruleset(s) major version", "Increment ruleset(s) minor version", and "Replace ruleset(s) version". At the bottom, there are three buttons: "Cancel", "Previous", and "Next".

13. Select the RES\_IIB server (this has already been created for you).  
Click Deploy.



The screenshot shows a configuration dialog titled "Select Server". At the top, there are five tabs: "Compose", "Query", "Analyze", "Project", and "Configure", with "Configure" being the active tab. Below the tabs, the breadcrumb path is "Deployment Baseline > RuleApp target > Versioning Policy > Select Server". The main content area has the heading "Select Server" and a text prompt: "Select the server where you want to deploy the RuleApp". To the right of the prompt is a dropdown menu with "RES\_IIB" selected. At the bottom, there are three buttons: "Cancel", "Previous", and "Deploy".

- After a few seconds, the deployment should have completed. You should see something like this:

**Deployment Succeeded**

**RuleApp archive deployed** ODMEmployeeBonus  
**Versioning Policy** Increment RuleApp major version  
**Rule Execution Server** http://localhost:9080/res

Archive content	Operation	Result
/ODMEmployeeBonus/1.0	Element added	/ODMEmployeeBonus/1.0
/EmployeeBonus/1.0	Element added	/EmployeeBonus/1.0

- To check it has deployed successfully, login to Rule Execution Server Console (open a new tab on the Firefox browser; and use the supplied bookmark in the ODM folder).

Login with the user resAdmin, password resAdmin.

Click Explorer.

**IBM Rule Execution Server console**

Home Explorer Decision Warehouse Diagnostics Server Info REST API

Home

**Welcome to the Rule Execution Server console**

- Explorer** Use the Explorer to deploy, browse, and modify RuleApps.
- Decision Warehouse Search and view decision traces.
- Diagnostics Run the server diagnostics to verify installation.
- Server Info View server configuration information and logged events.
- REST API Access the test tool for the resource management REST API. Use this tool to format and send requests, and to view responses.



16. You will see the new RuleApp.

The screenshot shows the IBM Rule Execution Server console. The top navigation bar includes 'Home', 'Explorer', 'Decision Warehouse', 'Diagnostics', 'Server Info', and 'REST API'. The 'Explorer' tab is active, showing a breadcrumb 'Explorer > RuleApps'. On the left, a 'Navigator' pane shows a tree structure with 'RuleApps (1)', 'Resources', 'Libraries', and 'Decision Services'. The main content area is titled 'RuleApps View' and contains three buttons: 'Add RuleApp', 'Deploy RuleApp Archive', and 'Update RuleApps'. Below these buttons, a summary box shows 'Total Number of RuleApps 1'. A table below that lists the RuleApps:

Select All	Name	Version	Creation Date	Number of rulesets
<input type="checkbox"/>	ODMEmployeeBonus	1.0	Dec 12, 2014 8:47:02 AM GMT-05:00	1

Below the table, it says 'RuleApp 1 - 1 of 1' and 'prev 10 next 10'. There is also a 'View only' dropdown set to 'Latest'.

The rule has now been fully deployed to the Rule Execution Server, and is ready for runtime use.

You are now ready to configure the Integration Bus components of this scenario.

## 3. Configure the Integration Bus Application with the ODM Rule

### 3.1 Import the Rule into the Integration Bus Toolkit

The business rule that you have defined in ODM will be invoked from the EmployeeService integration service that is defined in Integration Bus. To do this, you will extend the service to add a new operation, checkBonus. This new operation will incorporate a Decision Service node which will invoke the rule from the ODM Rule Execution Server.

1. First, to ensure there are no conflicts with other components, create a new IIB workspace in the IIB Toolkit, for example name it c:\workspaces\IIB\_ODM.

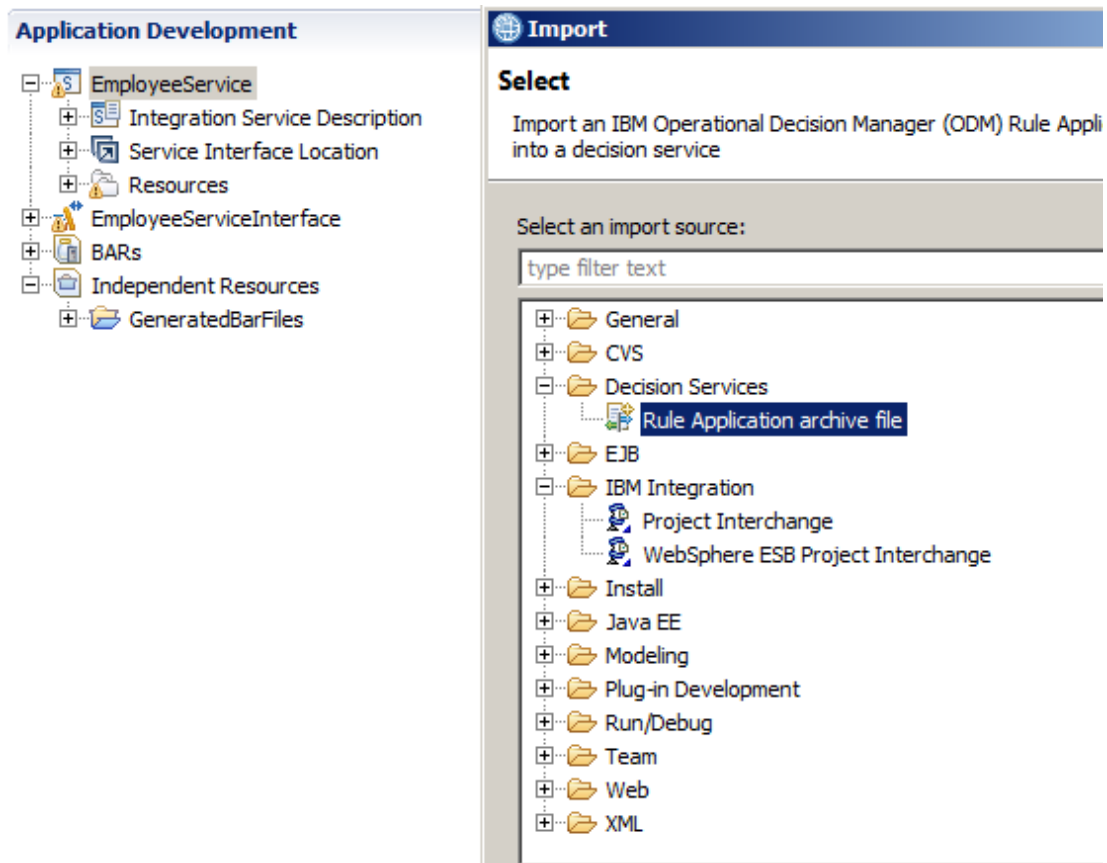
In the IIB Toolkit, import the Project Interchange files:

- c:\student10\Integration\_service\solution\EmployeeServiceInterface.V10.zip
- c:\student10\Integration\_service\solution\EmployeeService.V10.zip

Import all available projects in these PI files.

2. Right-click the EmployeeService project and click Import, and expand Decision Services.

Select Rule Application archive file. Click Next.

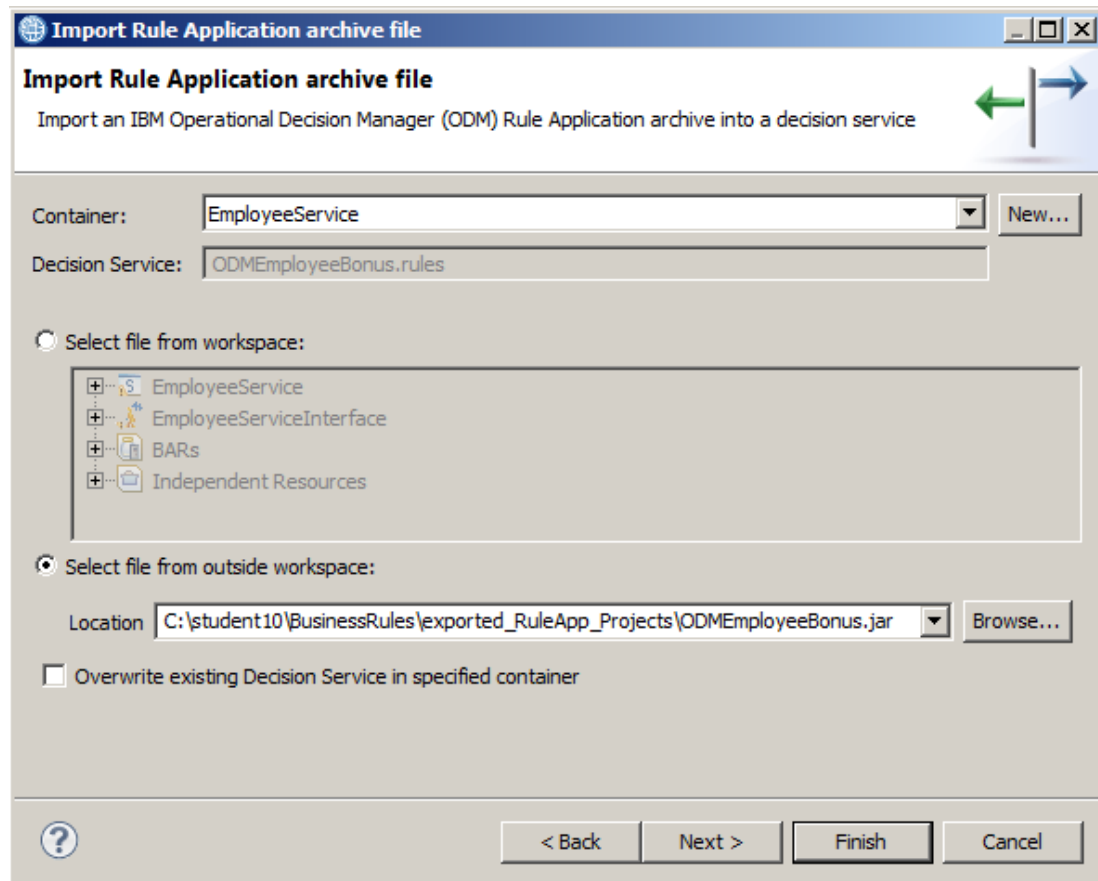


3. Select "File from outside workspace", and navigate to:

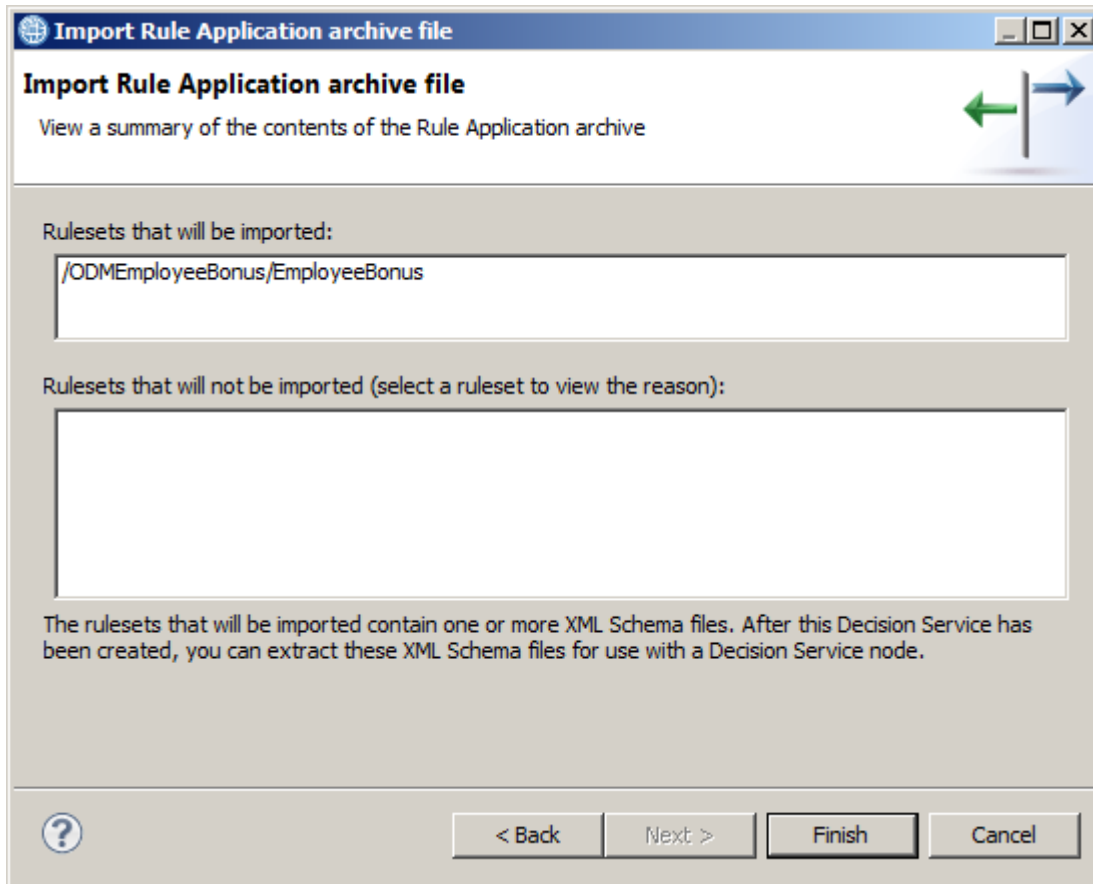
c:\student10\BusinessRules\exported\_RuleApp\_Projects\ODMEmployeeBonus.jar.

(This is the rule that you created earlier in the ODM Rule Designer).

Click Next.

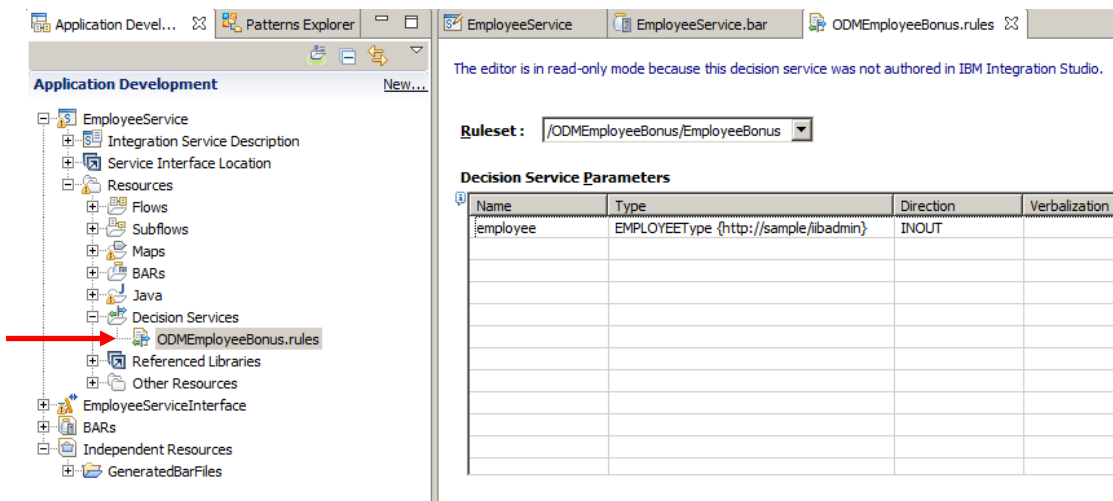


- At the summary window, click Finish.



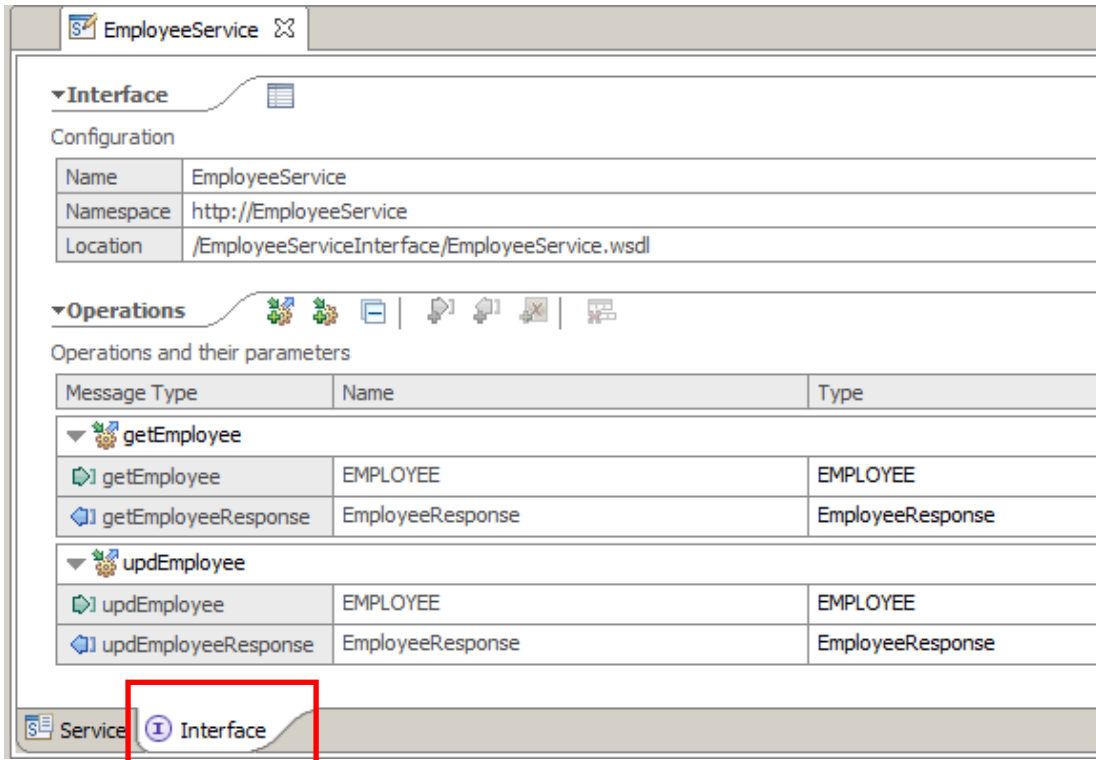
- When the rule archive has been imported, it will appear in the EmployeeService project.

Note that the rule cannot be edited in the Integration Bus Toolkit, and the verbalization will not be shown, since it was created by the ODM Rule Designer.



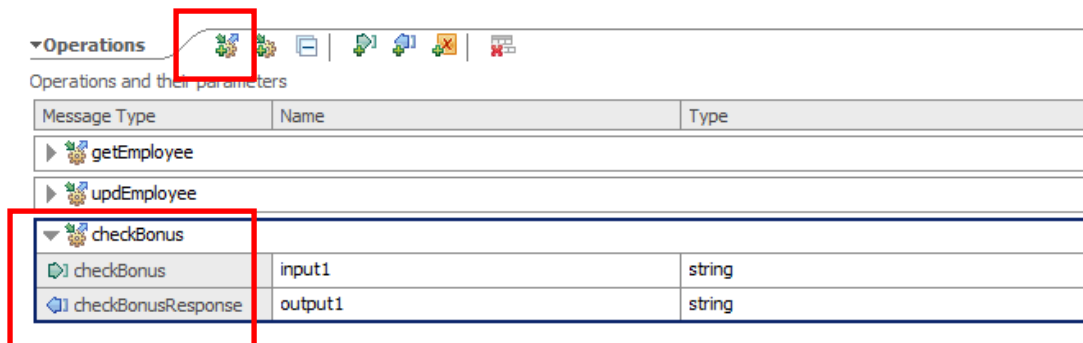
- Return to the EmployeeService integration service (if it's not currently open, reopen by double-clicking the Integration Service Description in the navigator).

Select the Interface tab. This will show the getEmp and updEmp operations.



- Click the Add Operation button which will create a new operation called "operation1". Overtyping this to name the new operation "checkBonus".

Press return, which will automatically change the names of the input and output message types (checkBonus and checkBonusResponse).



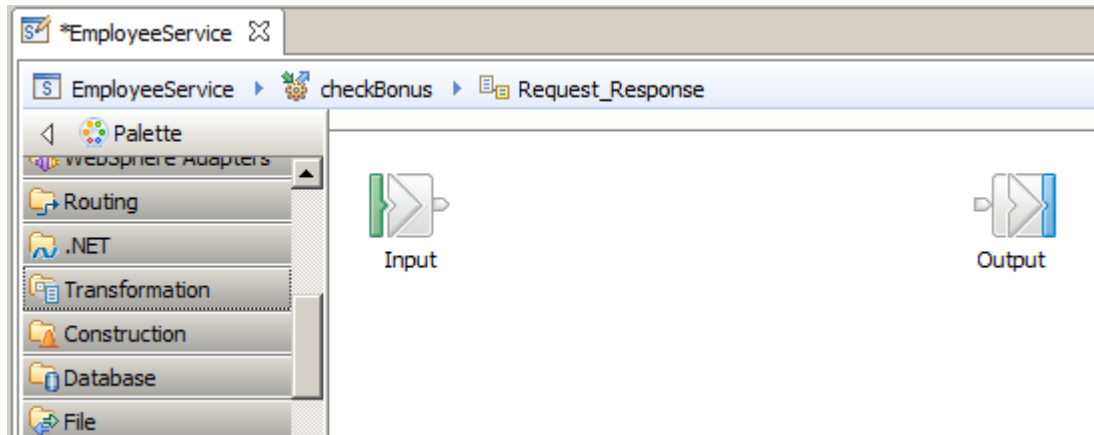
- Click the word "string" to change the type of both the input and output messages to EMPLOYEE. Use the Browse button when the dialogue window opens, and type "e" to show all available types that beginning with "e".

Save the changes to the service.

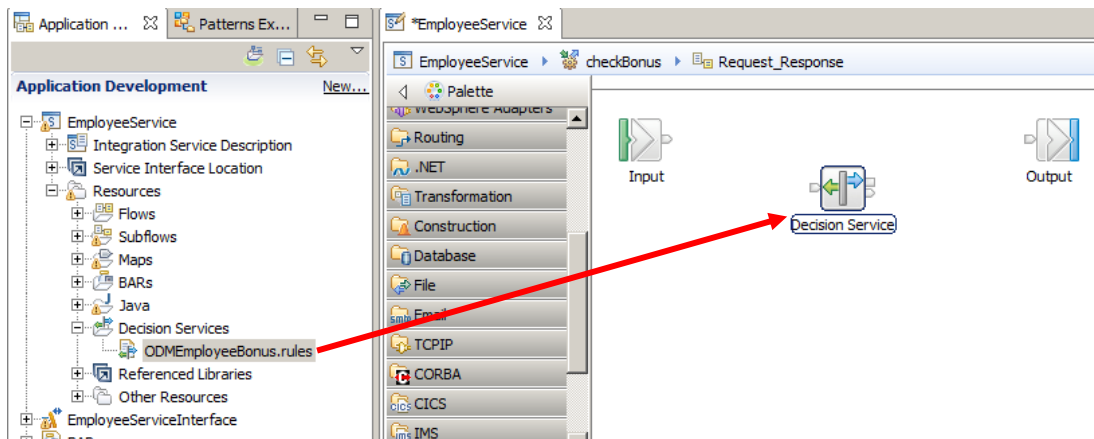
Message Type	Name	Type
▼ getEmployee		
getEmployee	EMPLOYEE	EMPLOYEE
getEmployeeResponse	EmployeeResponse	EmployeeResponse
▼ updEmployee		
updEmployee	EMPLOYEE	EMPLOYEE
updEmployeeResponse	EmployeeResponse	EmployeeResponse
▼ checkBonus		
checkBonus	EMPLOYEE	EMPLOYEE
checkBonusResponse	EMPLOYEE	EMPLOYEE

- Click the Service tab to show the implementation. Note that checkBonus is greyed out, since it has not yet been implemented.

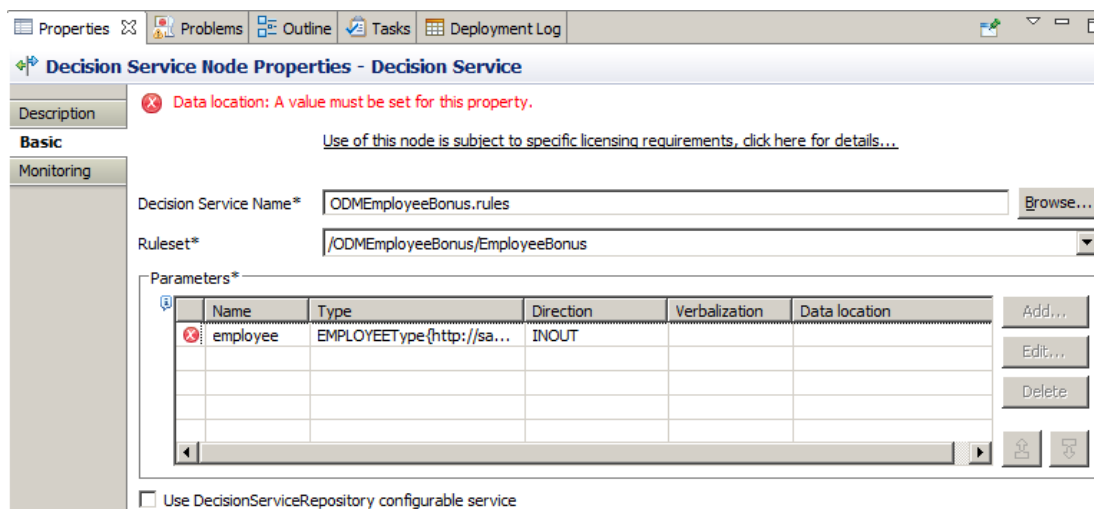
- Click checkBonus, which will open the subflow editor.



- Drop the ODMEmployeeBonus.rules decision service onto the flow editor.



- Note that the properties of the Decision Service node will show an error. You must specify the ruleset parameters (ie. the data location) in the input message tree, to allow the decision service node to locate the data that it will operate on.



13. Highlight the "employee" parameter and click the Edit button to open the XPath Expression builder.

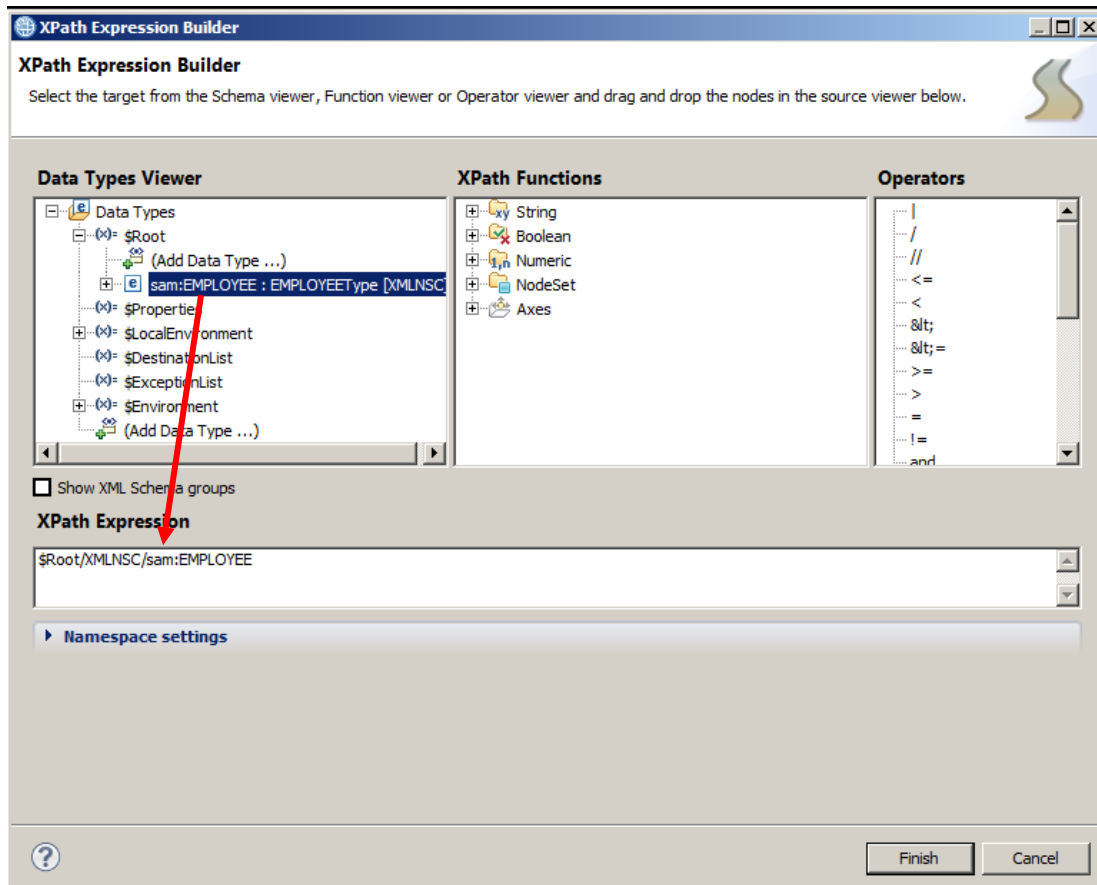
Click Edit in the pop-up window.

You may need to add a new Data Type (EMPLOYEE), under \$Root. Do this by expanding \$Root, clicking (Add Data Type...), and choose EMPLOYEE. Make sure this is added with the XMLNSC Domain.

Drag/drop the resulting EMPLOYEE data type onto the XPath Expression.

The final XPath expression should be **\$Root/XMLNSC/sam:EMPLOYEE**

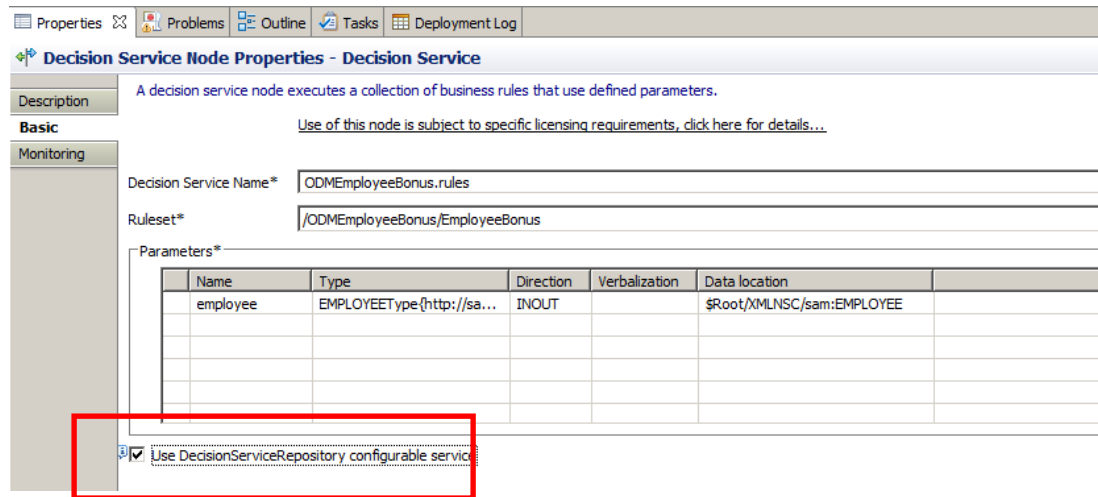
Click Finish, then OK to close the XPath wizards.





#### 14. Don't forget this next bit – it's important !

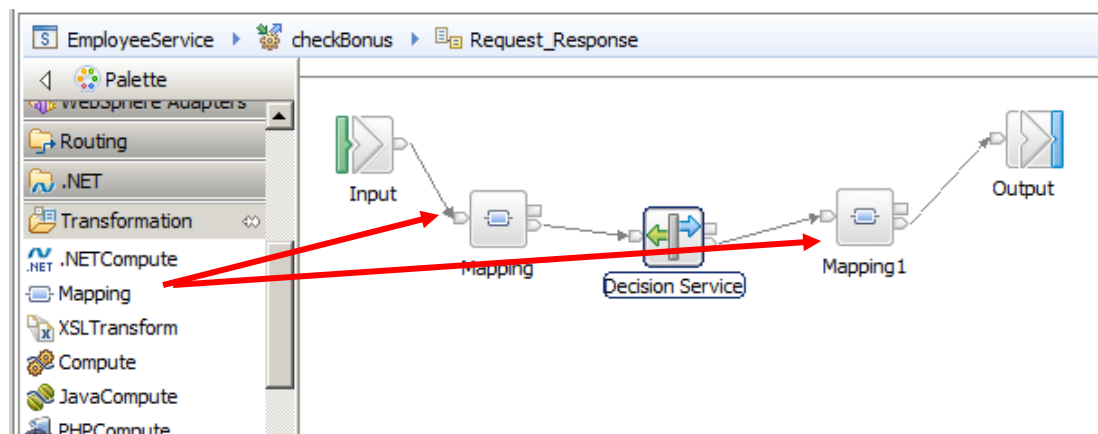
Select the option to use the DecisionServiceRepository configurable service. This will be used by the Integration Bus node to locate and load the EmployeeBonus ruleset from the Rule Execution Server database. If you don't do this, Integration Bus will attempt to locate a locally-defined rule in IIB, not a rule located in the external ODM database.



15. The new operation must make sure that the data that is sent to the Decision Service is in the correct format. The rule was defined to operate on the EMPLOYEE schema, so the checkBonus operation must send the data in this format.

To do this, a Mapping node will be added to transform the incoming message from the checkBonus SOAP message format to the EMPLOYEE XML schema format. A second Mapping node will be added to perform the reverse transformation.

In the checkBonus operation, add two mapping nodes, and connect them as shown.



Save the flow at this point (ctrl-S).

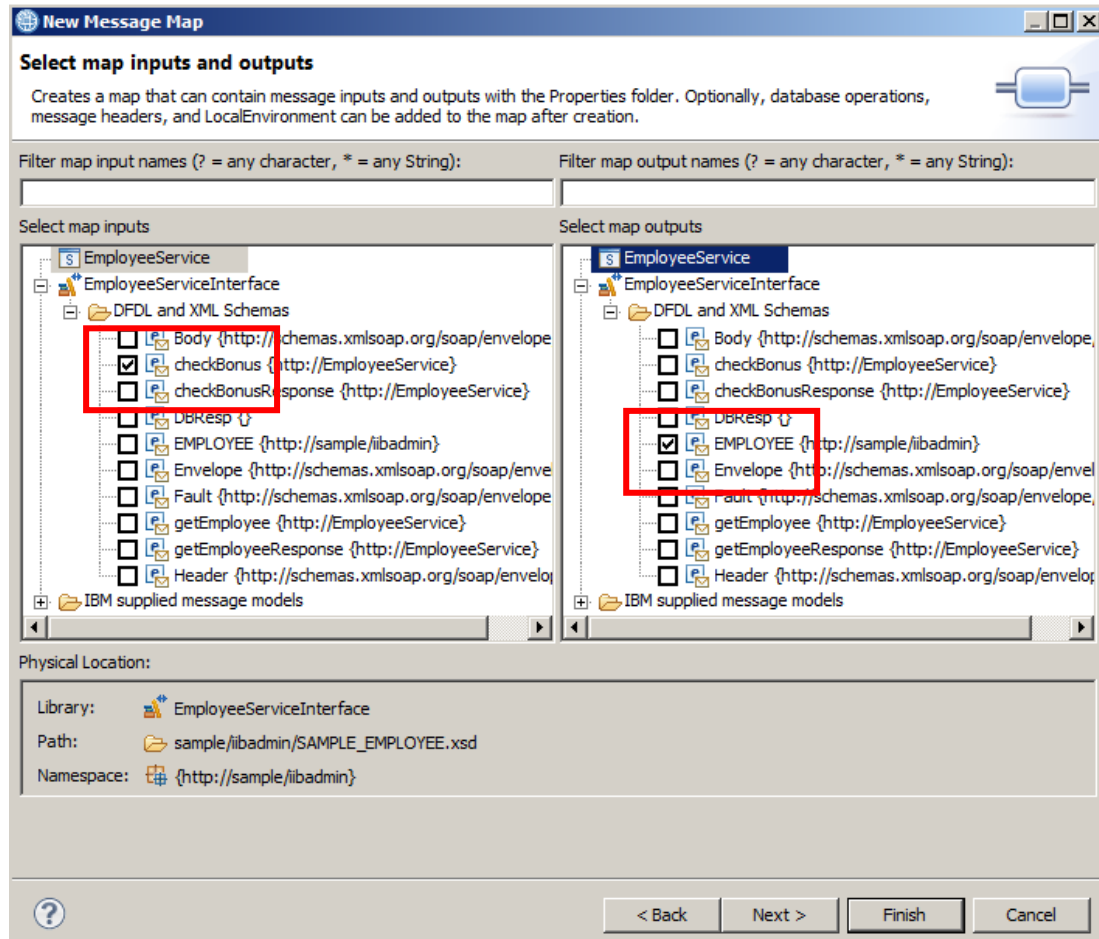
16. Open the first mapping node, and click Next on the first dialogue window.

At the map inputs and outputs window:

- Select the input message = checkBonus
- Select the output message = EMPLOYEE

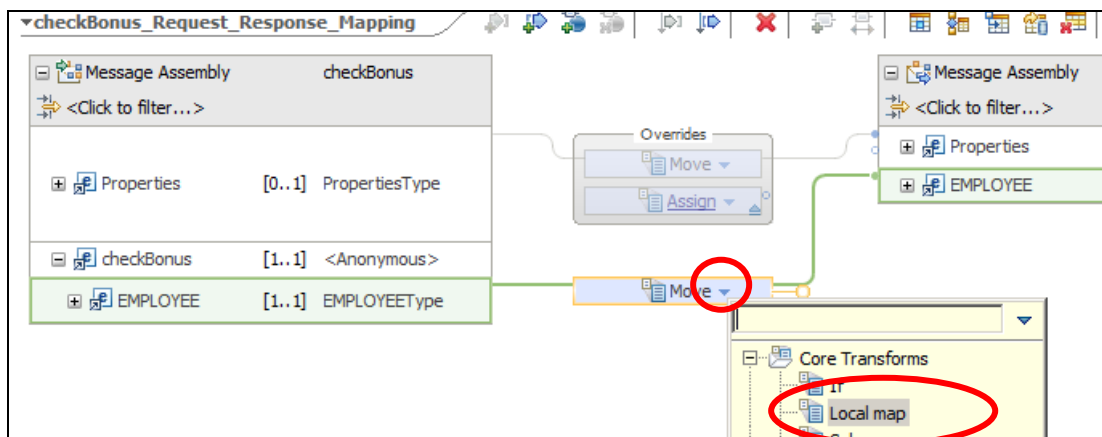
(If you don't see checkBonus in the available schemas for the map, it may be because you haven't saved the integration service - quit this map, save the service, and reopen the map).

Click Finish.



17. Expand checkBonus, and map the input EMPLOYEE to the output EMPLOYEEE.

A Move transform will be created; change the transform type to "Local map".



Then click "Local Map" to specify the individual element mappings. Use the AutoMap button to map each element (since the element names are all the same, just accept the default mappings).

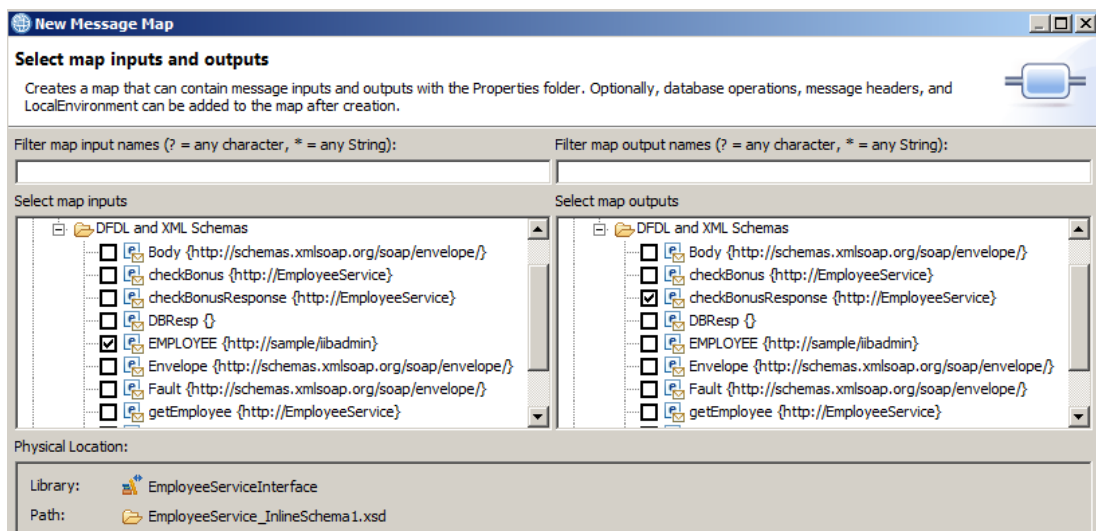
Save and close the first map.

18. Open the second map, and click Next on the first dialogue window.

At the map inputs and outputs window:

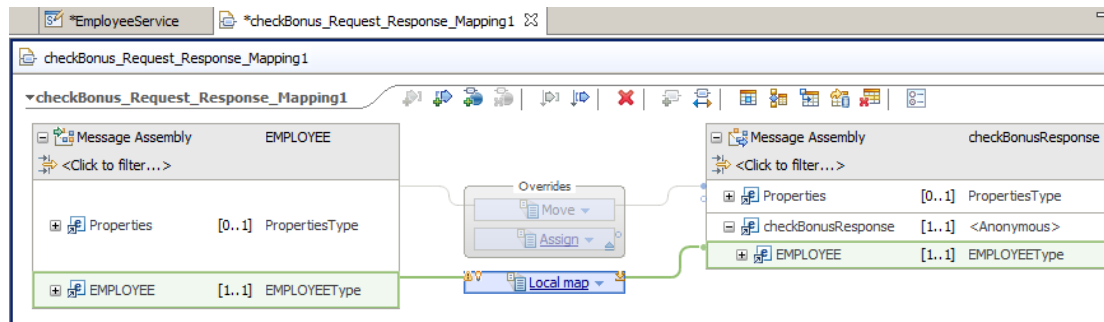
- Select the input message = EMPLOYEE
- Select the output message = checkBonusResponse

Click Finish.



19. In the output message assembly, expand checkBonusResponse, and map the input EMPLOYEE to the output EMPLOYEE.

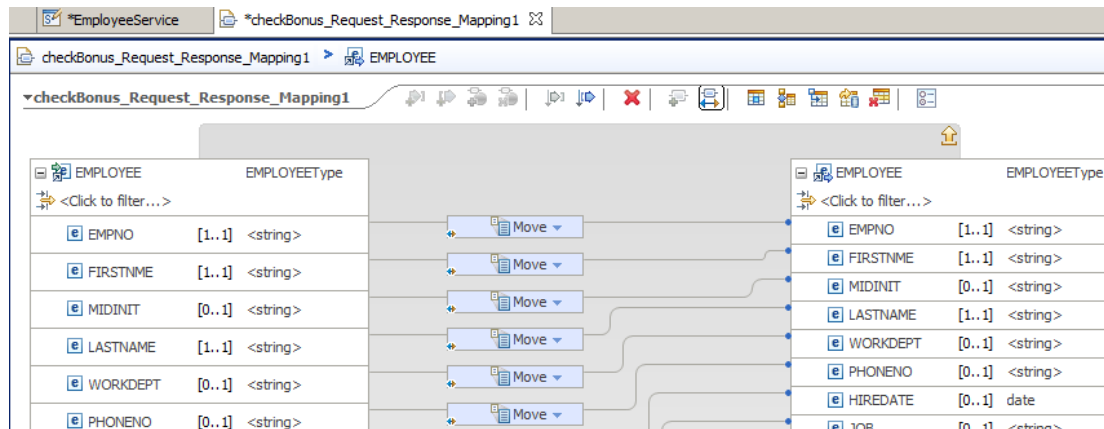
Change the transform type to "Local map".



20. Click "Local map" to specify the individual mappings.

Use the Automap button to map each input element to the same output element.

Save and close the map.



### 3.1.1 Review the Integration Bus Configurable Services for Decision Services

To enable connectivity to the desired ODM Rule Execution Server, two configurable services have to be defined.

The first one defines a JDBC connection to the underlying DB2 database for ODM. The second one is the DecisionServices configurable service, which references the JDBC configurable service.

These configurable services have already been defined for the IB10NODE Integration Bus workshop environment, using a set of supplied scripts. We will review these definitions using the web browser administration tools for IIB.

1. In the Firefox browser, use the supplied bookmark in the IIB folder to connect to IB10NODE, port 4421 (https).

Expand IB10NODE, Operational Policy, Configurable Services, JDBC Providers.

Select the **ODMDS** JDBC Provider configurable service.

Note the following properties have been changed from the default values:

- portNumber = 50000
- serverName = localhost
- jarsURL = c:\IBM\SQLLIB\java
- databaseName = ODMDS
- databaseVersion = 10.1
- securityIdentity = ODMRESecurityId

This configurable service can be defined or changed through the web admin interface if required. However, do not make any changes to this definition.

The screenshot shows the web administration interface for the ODMDS - JDBCProviders Configurable Service. The left sidebar contains a tree view of services, with 'JDBCProviders' expanded to show 'ODMDS'. The main content area displays the 'Overview' tab with a 'Properties' section containing the following configuration details:

Property Name	Value
type4DatasourceClassName	com.ibm.db2.jcc.DB2XADataSource
type4DriverClassName	com.ibm.db2.jcc.DB2Driver
databaseType	DB2 Universal Database
jdbcProviderXASupport	true
portNumber	50000
connectionUriFormatAttr5	
connectionUriFormatAttr4	
serverName	localhost
connectionUriFormatAttr3	
connectionUriFormatAttr2	
connectionUriFormatAttr1	
environmentParms	default_none
maxConnectionPoolSize	0
description	default_Description
jarsURL	C:\IBM\SQLLIB\java
databaseName	ODMDS
databaseVersion	10.1
securityIdentity	ODMRESecurityId
connectionUriFormat	jdbc:db2://[serverName]:[portNumber]/password:

2. Switch to the DecisionServiceRepository configurable service category.

Select the Default configurable service.

The following values have been set:

- resManagementPort = 1883 (matches the Management Port in the ODM Rule Execution Server configuration. Additionally, the ODM Rule Execution Server must be configured to publish rule changes over TCP/IP. The default protocol is JMX, so this must be changed to "tcpip" in the Rule Execution Server. This has been done for you on the pre-built VMWare image.
- JDBCProviderName = ODMDS (matches the JDBC definition shown above)
- resManagementHost = BETAWORKS-ESB10 (hostname of the ODM Rule Execution Server)

Do not make any changes to these definitions.

The screenshot displays the IBM Integration Bus Administration Console. The left-hand navigation pane shows a tree view of service categories, with 'DecisionServiceRepository' expanded and 'Default' selected. The main content area is titled 'Default - DecisionServiceRepository Configurable Service' and features an 'Overview' tab. Below the tab, a 'Properties' section is visible, containing a table with the following configuration details:

resManagementProfiling	false
resManagementPort	1883
JDBCProviderName	ODMDS
resManagementHost	BETAWORKS-ESB10

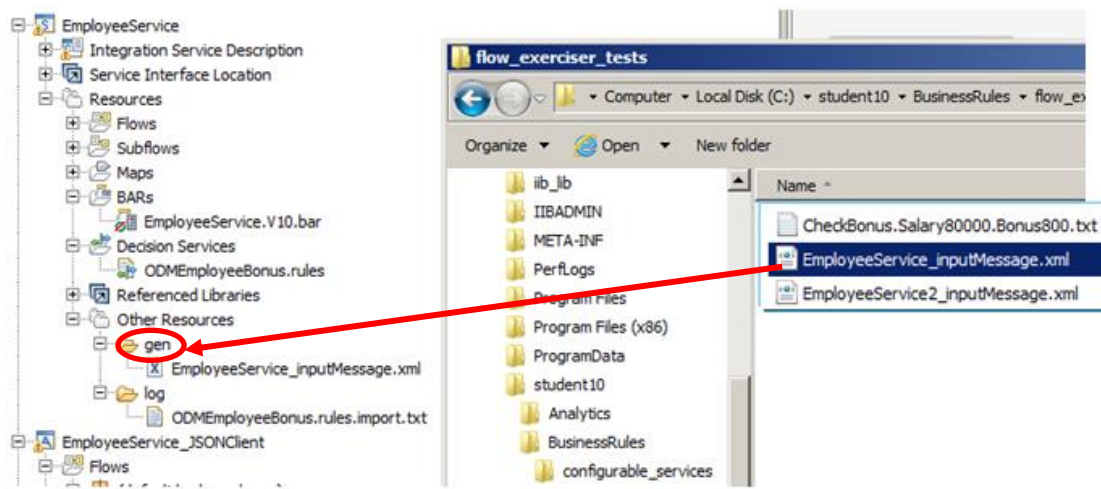
## 4. Test the service with the deployed rule

### 4.1 Test the service with the IIB Flow Exerciser

1. Because you built this new checkBonus operation on an existing EmployeeService, you will already have a pre-built version of some Flow Exerciser tests. These are contained in a file called EmployeeService\_inputMessage.xml. This is located under the "Other Resources / gen" folder.

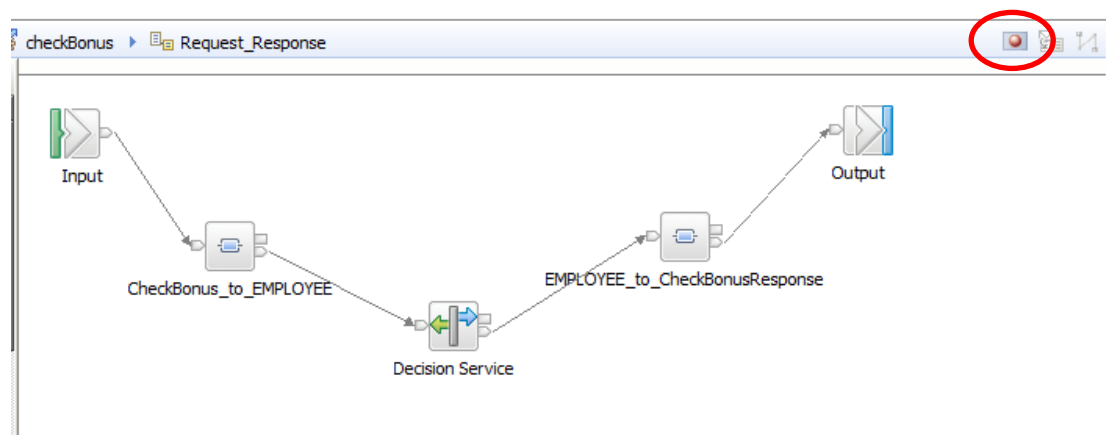
This lab provides a set of prebuilt tests for the checkBonus operation. Copy the file c:\student10\BusinessRules\flow\_exerciser\_tests\EmployeeService\_inputMessage.xml onto the "Other resources/gen" folder.

Confirm the file over-write.



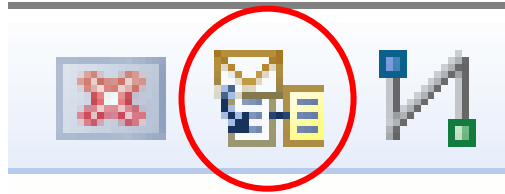
2. In the checkBonus operation, click the red Flow Exerciser button.

This will deploy the updated EmployeeService and the EmployeeServiceInterface library, and place the flow in test mode.





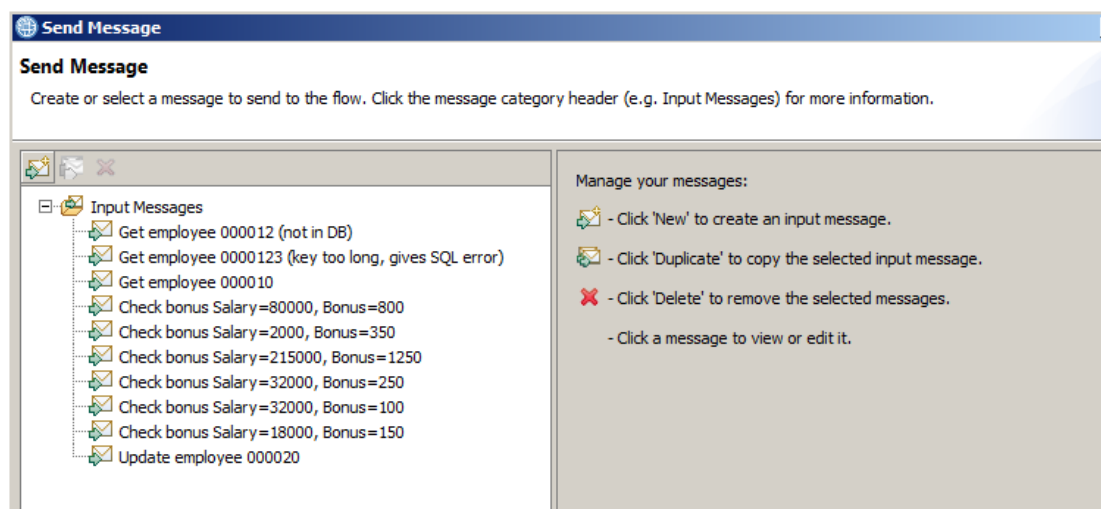
- When the flow editor has turned grey, click the Send Message icon.



- A number of test cases are provided.

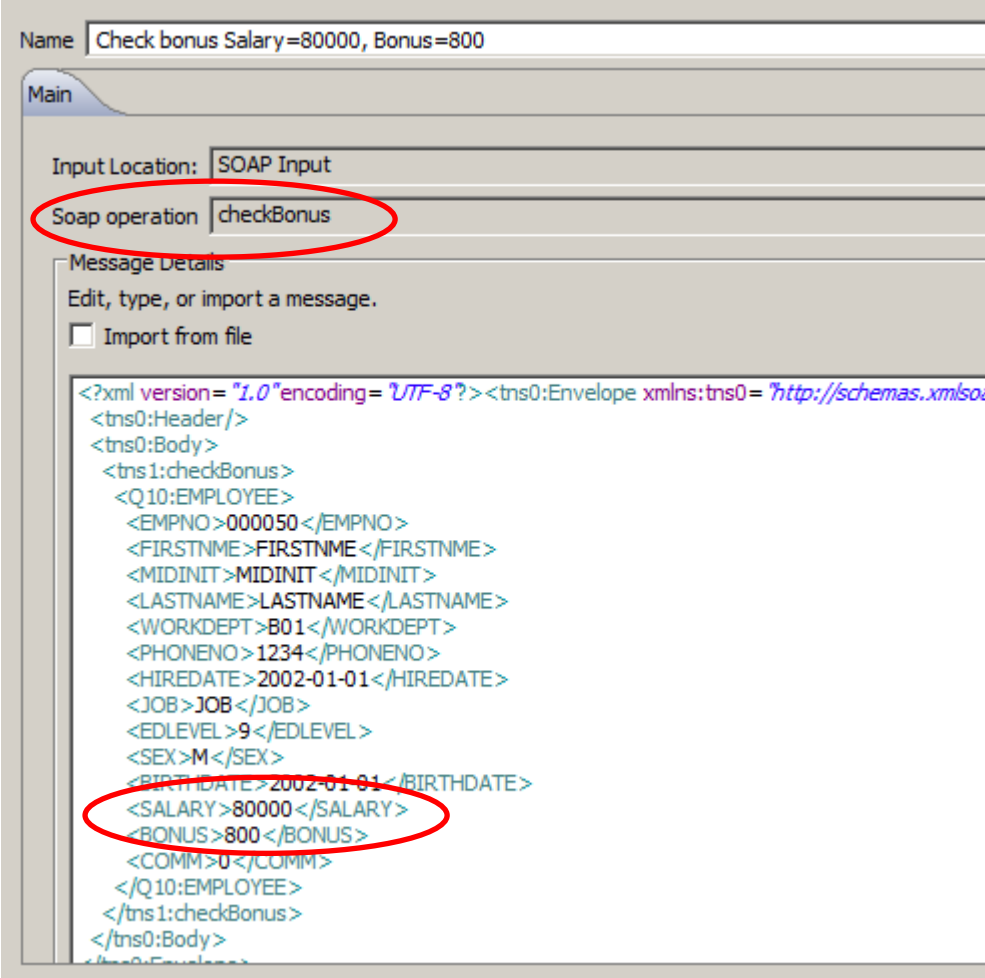
The test cases have been provided to test all three SOAP operations, and the operation value has been set to match the test case.

For this "Check Bonus" scenario, several combinations of values for Salary and Bonus are provided. Depending on the combination of these values sent to the checkBonus operation, it will either change the bonus (to 500), or leave the value unchanged.



5. Click the test case Salary=80000, Bonus=800.

You will see the two values specified in the body of the message. Also note that the SOAP operation is set to checkBonus.



Name: Check bonus Salary=80000, Bonus=800

Main

Input Location: SOAP Input

Soap operation: checkBonus

Message Details

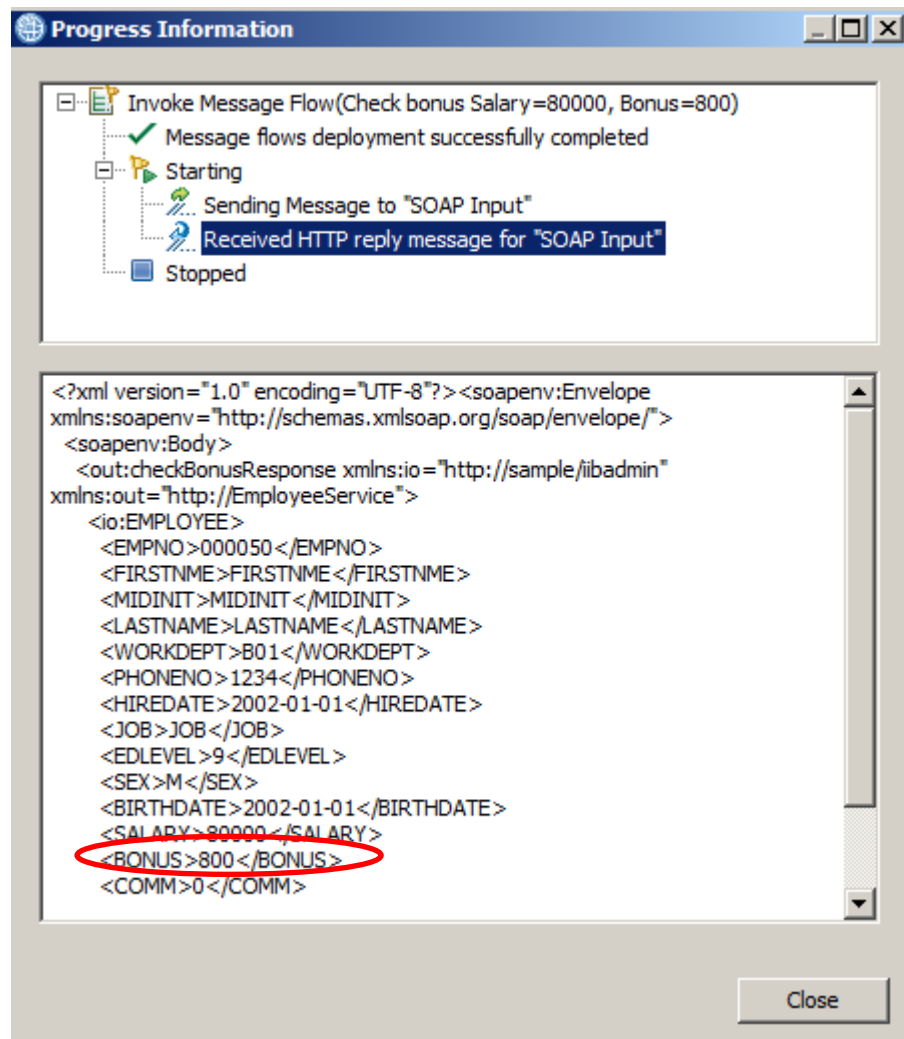
Edit, type, or import a message.

Import from file

```
<?xml version="1.0" encoding="UTF-8"?><tns0:Envelope xmlns:tns0="http://schemas.xmlsoap.org/soap/envelope/">
  <tns0:Header/>
  <tns0:Body>
    <tns1:checkBonus>
      <Q10:EMPLOYEE>
        <EMPNO>000050</EMPNO>
        <FIRSTNAME>FIRSTNAME</FIRSTNAME>
        <MIDINIT>MIDINIT</MIDINIT>
        <LASTNAME>LASTNAME</LASTNAME>
        <WORKDEPT>B01</WORKDEPT>
        <PHONENO>1234</PHONENO>
        <HIREDATE>2002-01-01</HIREDATE>
        <JOB>JOB</JOB>
        <EDLEVEL>9</EDLEVEL>
        <SEX>M</SEX>
        <BIRTHDATE>2002-01-01</BIRTHDATE>
        <SALARY>80000</SALARY>
        <BONUS>800</BONUS>
        <COMM>0</COMM>
      </Q10:EMPLOYEE>
    </tns1:checkBonus>
  </tns0:Body>
</tns0:Envelope>
```

## 6. Click Send.

The operation will execute, and pass the data to the ODM service. The defined rule will mean that the bonus will not be changed.



## 7. Run other tests from the CheckBonus set of tests. You will need to reopen the Send Message dialogue window to select further tests.

Depending on the combination of salary and bonus, the bonus value may or may not be updated with a new value. The test "Salary=18000, Bonus=150" will update the bonus to 500.

Note that you can also type new values into the flow test editor. And you can create new tests (right-click an existing test, and select Duplicate).

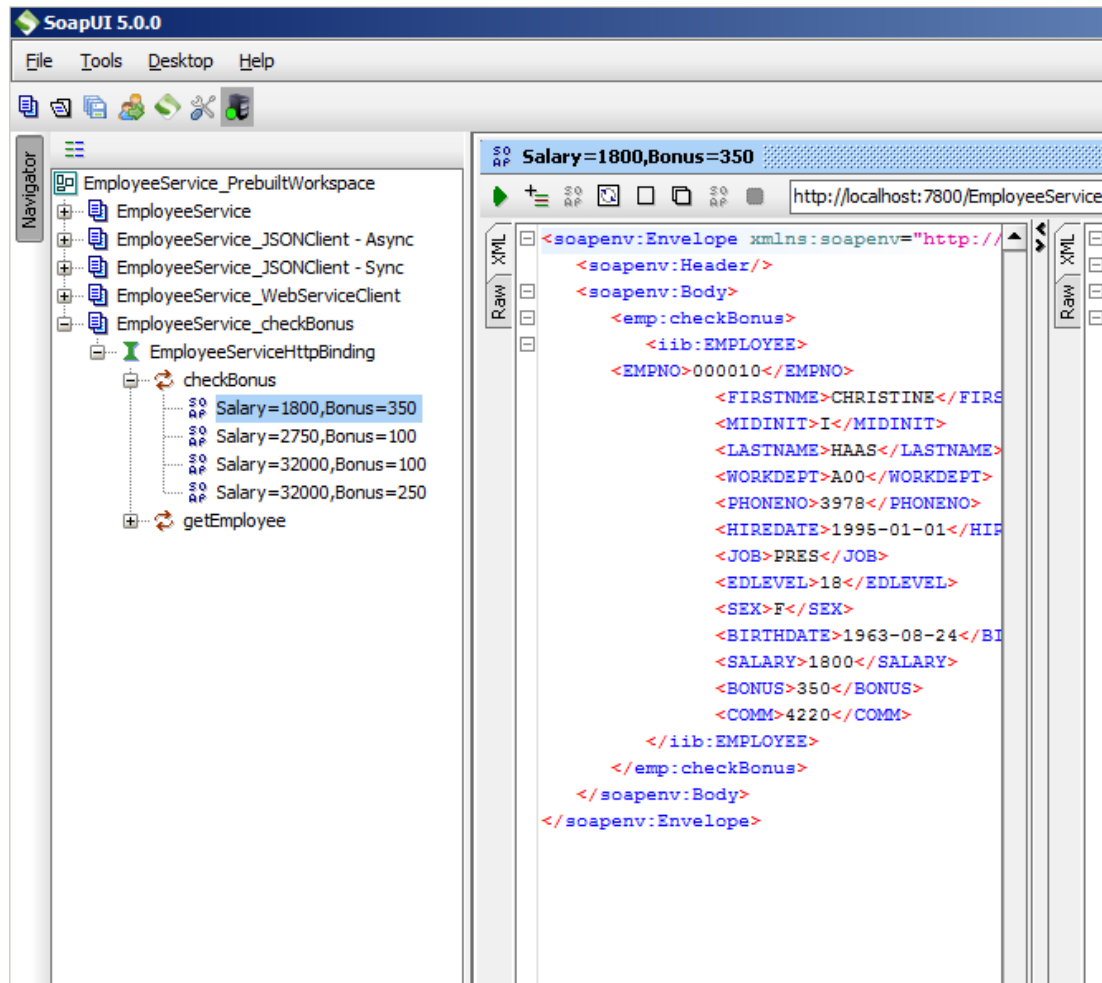
## 4.2 Test the service with SOAPUI (optional)

A similar set of tests are provided for use with SOAPUI. You can run these in combination with the Flow Exerciser (to see the path of execution), or without, to simply see the result.

1. From the Start menu, open SOAPUI.

Open the SOAPUI workspace EmployeeService\_PrebuiltWorkspace.

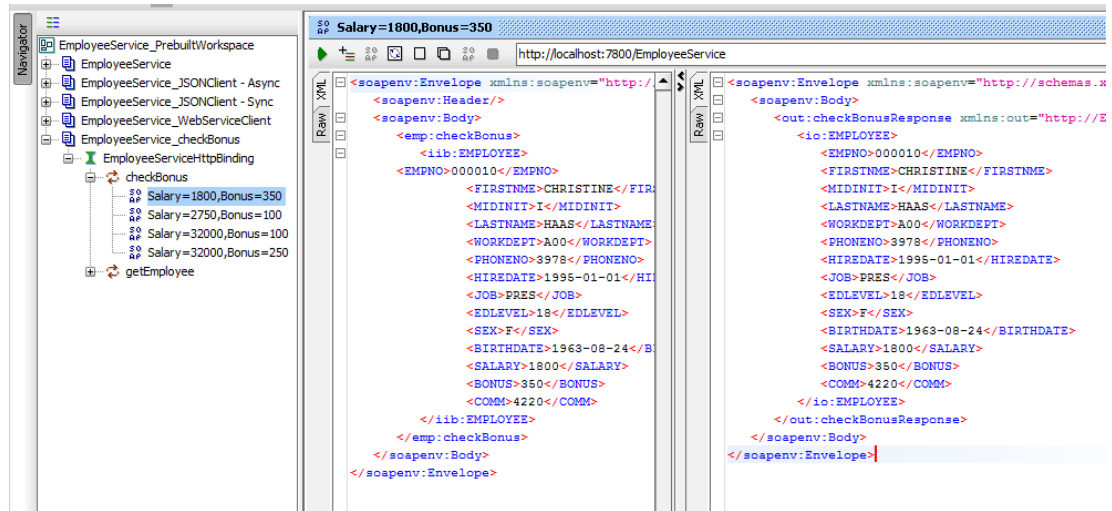
Expand the project EmployeeService\_checkBonus, then the checkBonus operation. You will see several input messages have been defined with various combinations of salary and current bonus.



2. Perform various tests with these input messages.

For example, the message with Salary=1800,Bonus=350 should not result in any change in the bonus value.

Click the green arrow to execute the checkBonus operation. You should see that the bonus is returned unchanged.



3. Use the input message with Salary=2750,Bonus=100. You should see that the bonus has been updated to 500.

## 5. Update the Business Rule

In this chapter you will:

- Update the new\_bonus rule using the Business Console
- Deploy the updated rule to the Rule Execution Server with the Decision Centre Enterprise Console

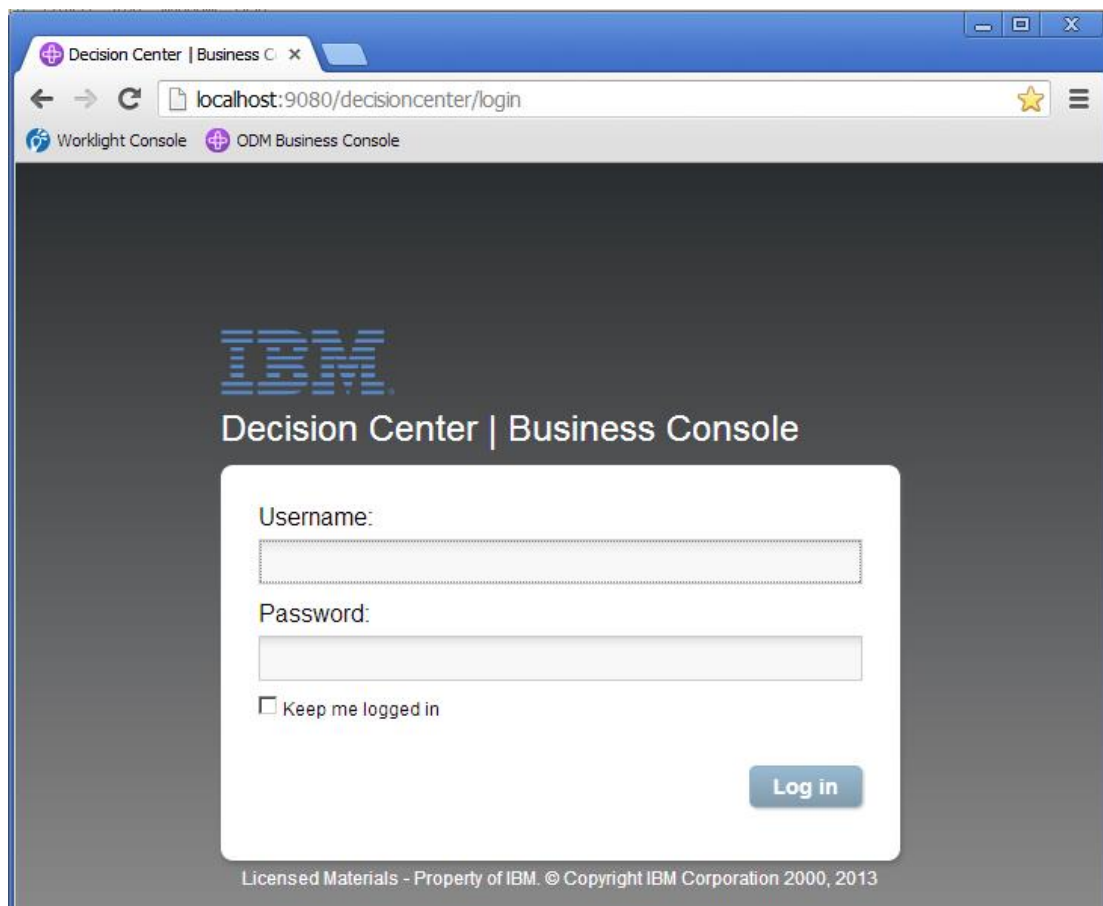
### 5.1 Update the Rule in the Business Console

The Decision Center Business Console is a browser-based tool that allows a business user to update and activate decision services (rules), subject to defined access controls.

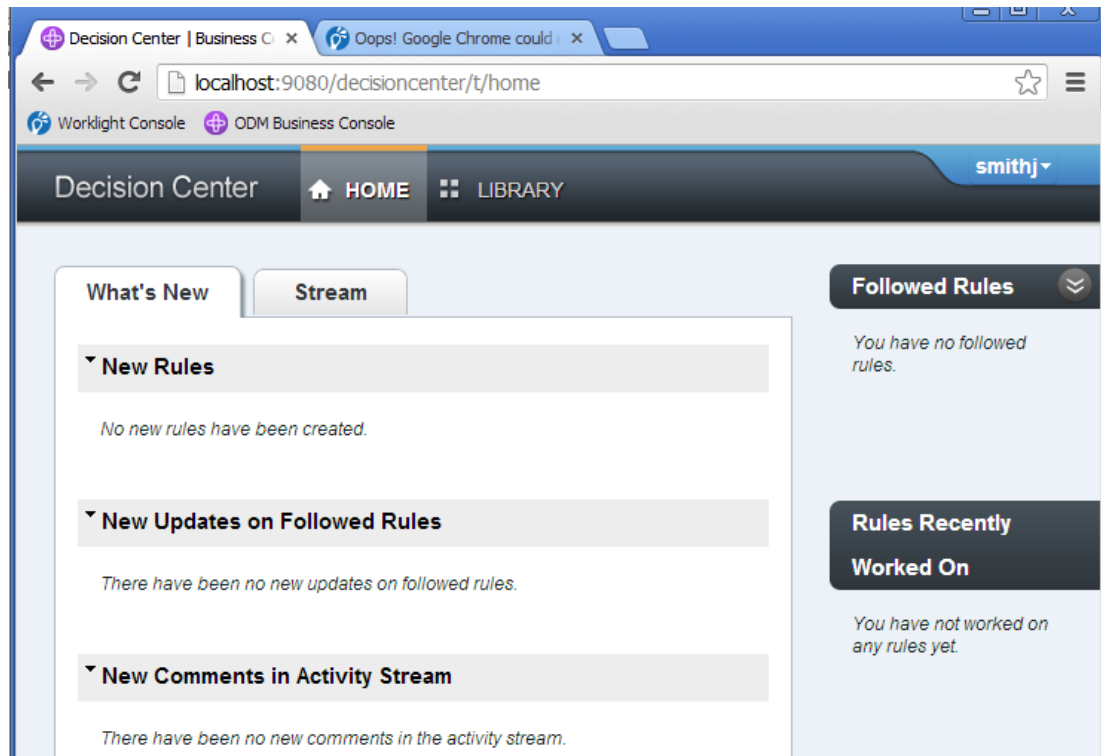
You will now login to the Business Console as a business user, and update the rule that was defined earlier. You will then retest the service to see that the new version of the rule has been used by the Integration Bus service.

1. Open a Chrome browser (to avoid confusion with other windows and users in Firefox), open the Decision Center Business Console using the provided bookmark (ODM Business Console).

Login as user “smithj” (password passw0rd).

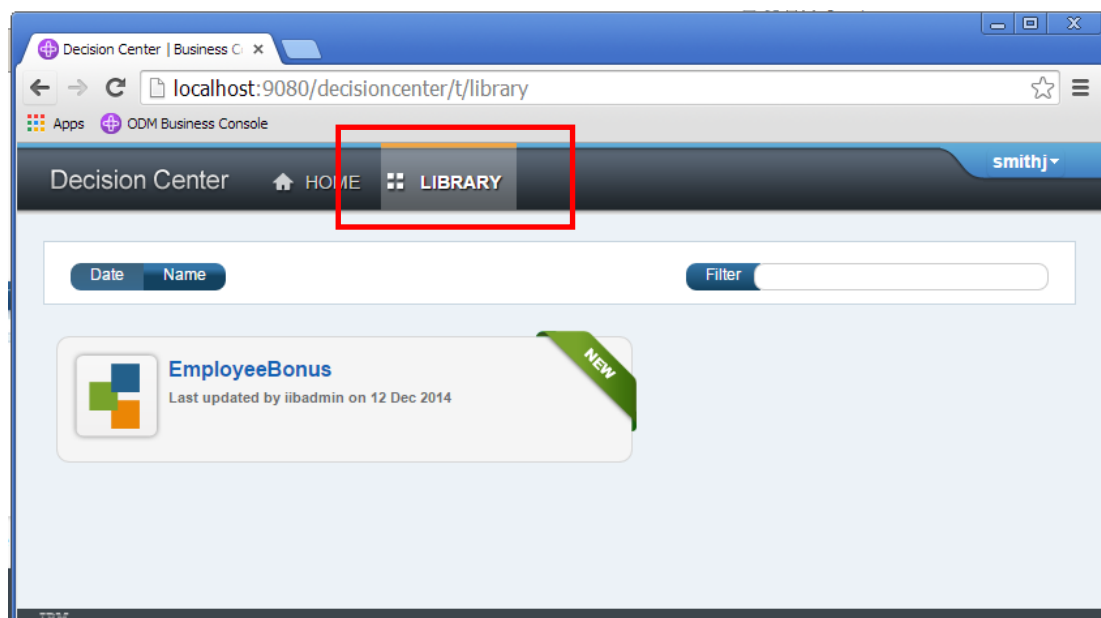


- Initially, this user does not have any projects that he is following.

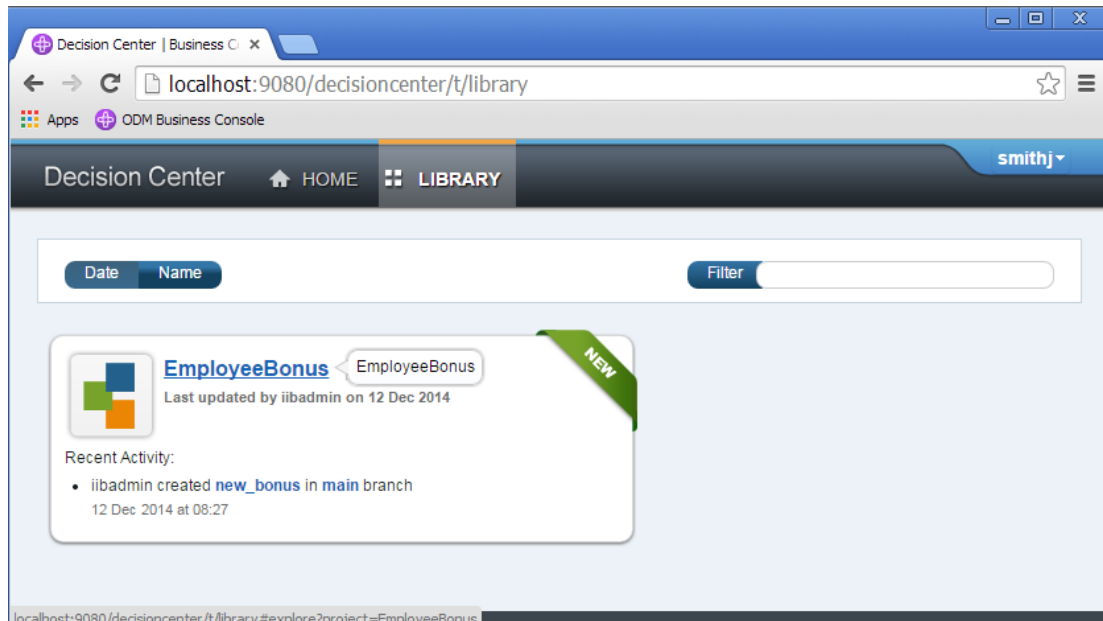


- Click Library.

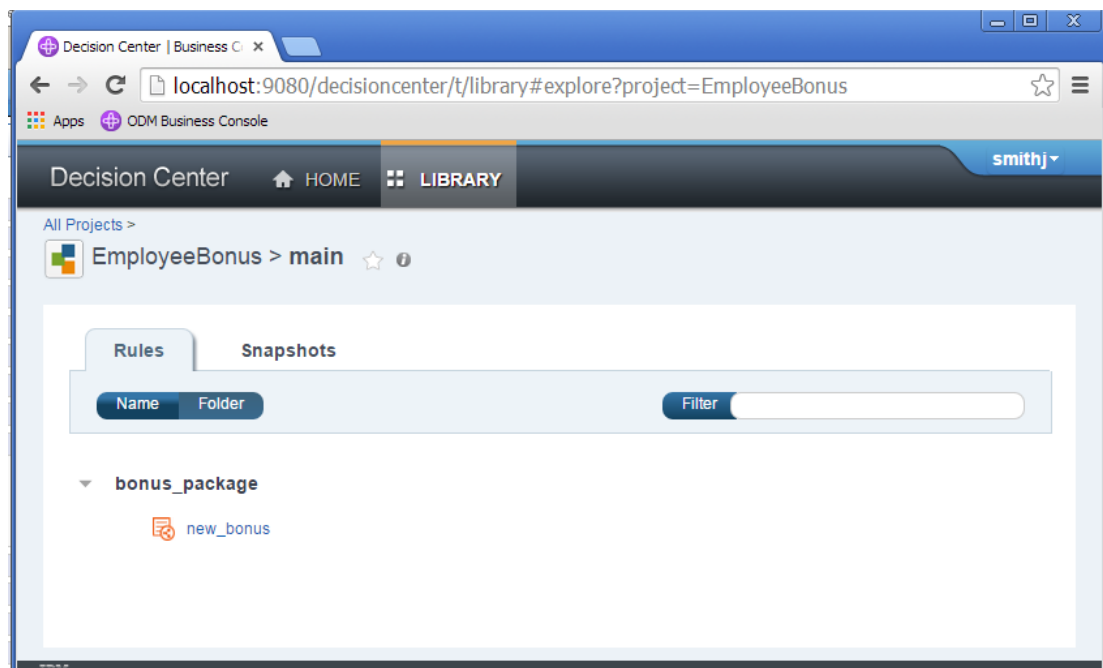
You will see that there is a project called EmployeeBonus, which is denoted as "New", showing the date that this was last updated.



4. Hover the mouse over the EmployeeBonus project. This will show additional details of the new project.



5. Now click the EmployeeBonus project.





- 6. Click the new\_bonus rule.

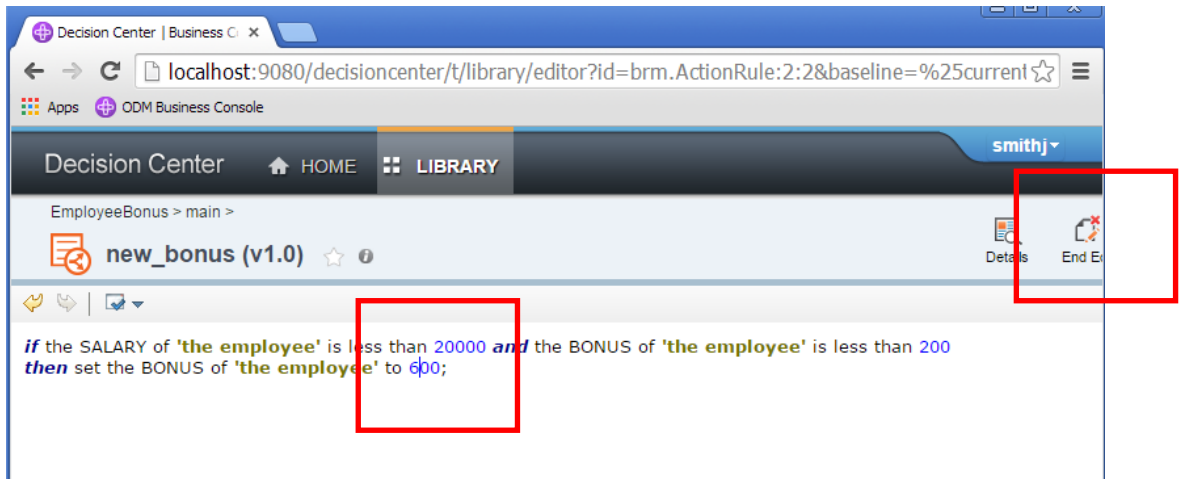
This will show the latest version of the rule, with the values that you originally entered with the Rule Designer.

Click Edit Rule.

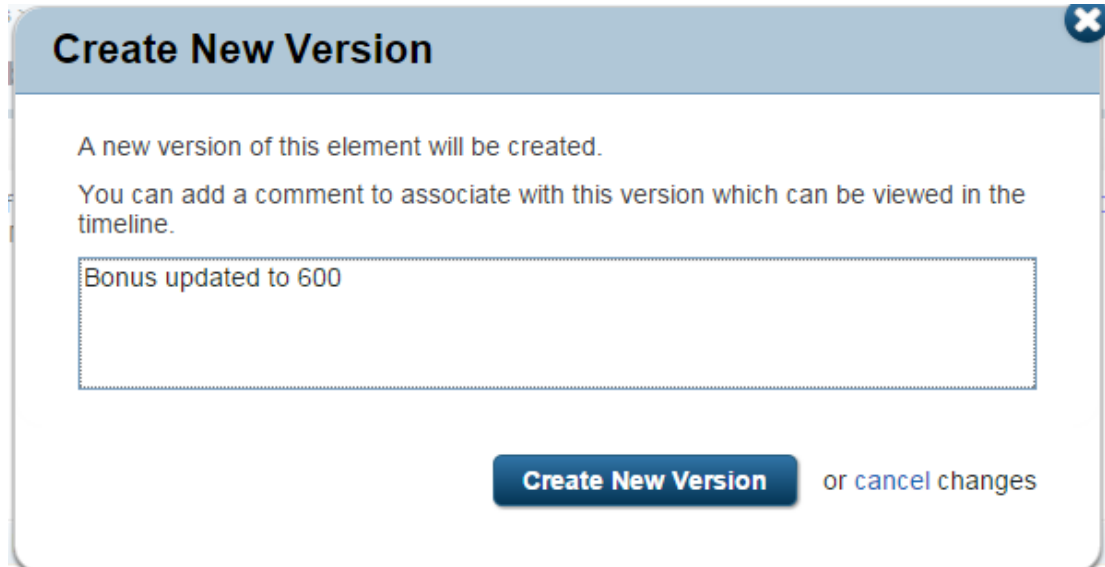


- 7. Change the value of the new bonus to a new value.

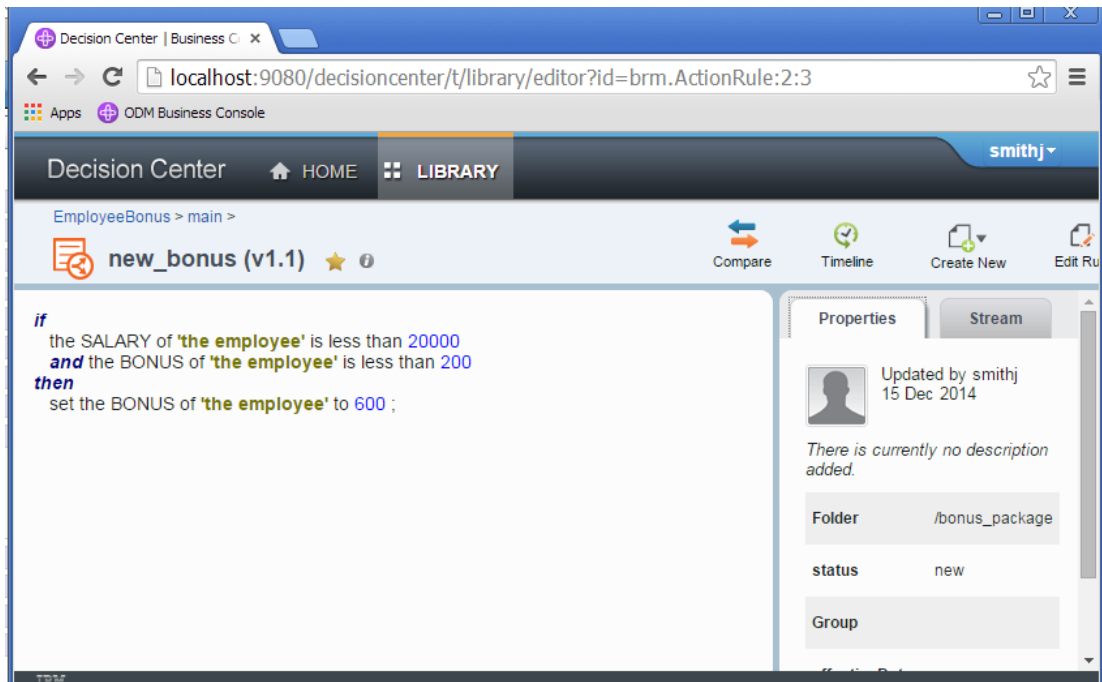
Click End Edit.



- This will create a new version of the rule. Provide a suitable comment, and click Create New Version.

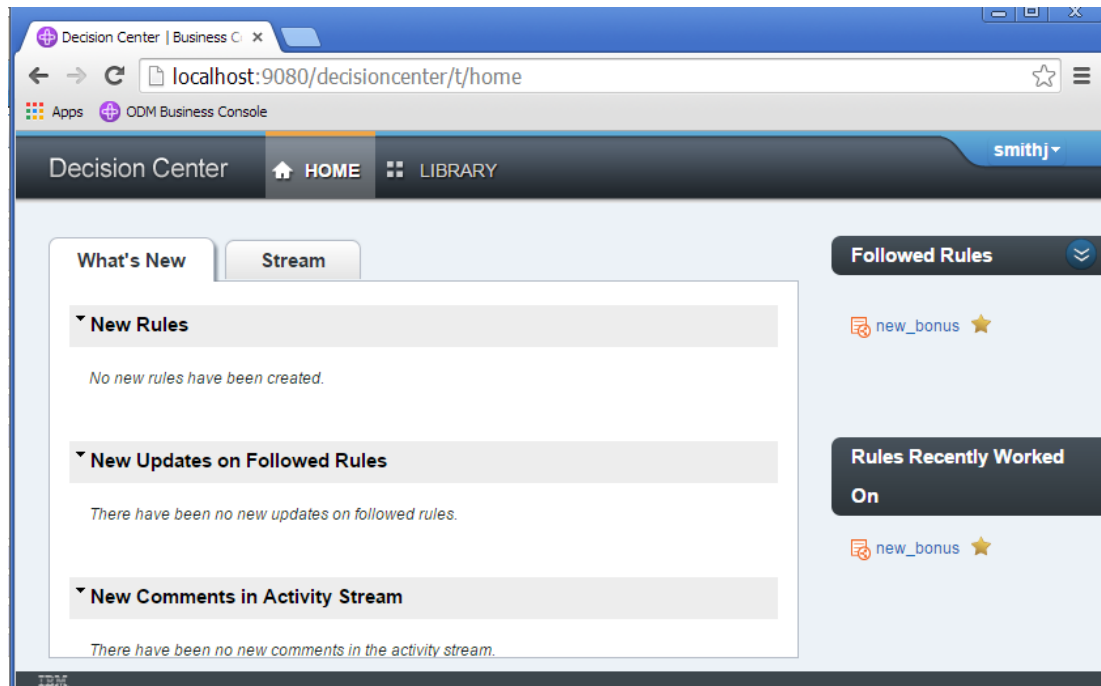


Note that a new version (v1.1) has been created, and change data descriptions have been recorded.



9. Click the Home tab.

Note that the new\_bonus rule has now become a “Followed Rule”, and it has also been added to the Recently Worked rules.



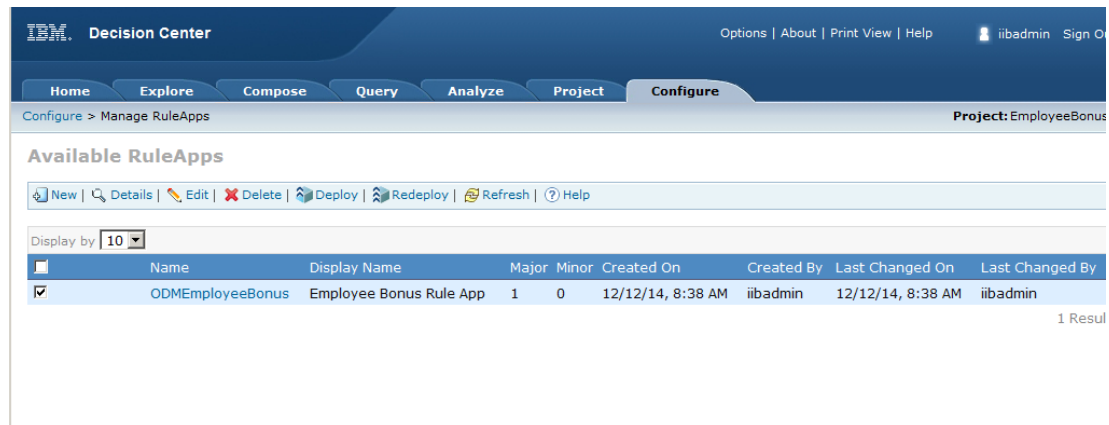
## 5.2 Use the ODM Enterprise Console to deploy the updated rule

Updated rules are deployed to the Rule Execution Server using the Decision Centre Enterprise Console.

1. Switch back to the Firefox browser. You may still be logged into the Enterprise Console, but if not, login now.

To redeploy the updated rule, click Configure, then click Manage RuleApps.

Select the ODMEmployeeBonus, and click Deploy.



IBM Decision Center

Options | About | Print View | Help | iibadmin | Sign Out

Home | Explore | Compose | Query | Analyze | Project | **Configure**

Configure > Manage RuleApps | Project: EmployeeBonus

**Available RuleApps**

New | Details | Edit | Delete | Deploy | Redeploy | Refresh | Help

Display by: 10

	Name	Display Name	Major	Minor	Created On	Created By	Last Changed On	Last Changed By
<input checked="" type="checkbox"/>	ODMEmployeeBonus	Employee Bonus Rule App	1	0	12/12/14, 8:38 AM	iibadmin	12/12/14, 8:38 AM	iibadmin

1 Result

2. Follow the same dialogue as the earlier deploy, but choose a RuleApp minor version upgrade when presented with the versioning policy.

Otherwise, click Next on all windows, and Deploy on the final one.

### Versioning Policy

- Increment RuleApp major version
- Increment RuleApp minor version
- Replace RuleApp version
- Increment ruleset(s) major version
- Increment ruleset(s) minor version
- Replace ruleset(s) version

- When complete, you will see that the version of ODMEmployeeBonus has been updated to version 1.1.

### Deployment Succeeded

**RuleApp archive deployed** ODMEmployeeBonus  
**Versioning Policy** Increment RuleApp minor version  
**Rule Execution Server** http://localhost:9080/res

Archive content	Operation	Result
/ODMEmployeeBonus/1.0	Version changed and element added	/ODMEmployeeBonus/ <b>1.1</b>
/EmployeeBonus/1.0	Version changed and element added	/EmployeeBonus/1.0

[Back](#)

- A quick check in the Rule Execution Server Console will show you the new version of the deployed rule (click RuleApps in the navigator to refresh the display).

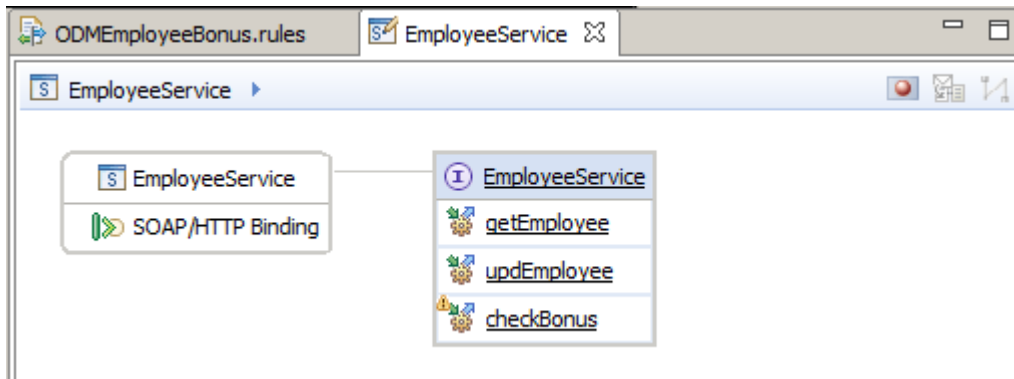
The screenshot shows the IBM Rule Execution Server console interface. The main area is titled "RuleApps View" and shows a list of RuleApps. The "Total Number of RuleApps" is 2. The list contains two entries:

Select All	Name	Version	Creation Date	Number of rulesets	Download Archive
<input type="checkbox"/>	ODMEmployeeBonus	1.0	Dec 12, 2014 8:47:02 AM GMT-05:00	1	
<input type="checkbox"/>	ODMEmployeeBonus	1.1	Dec 15, 2014 4:39:05 AM GMT-05:00	1	

The interface also includes a "Navigator" on the left with "RuleApps (2)" selected, and a "View only" dropdown set to "Latest version".

## 6. Test the service again

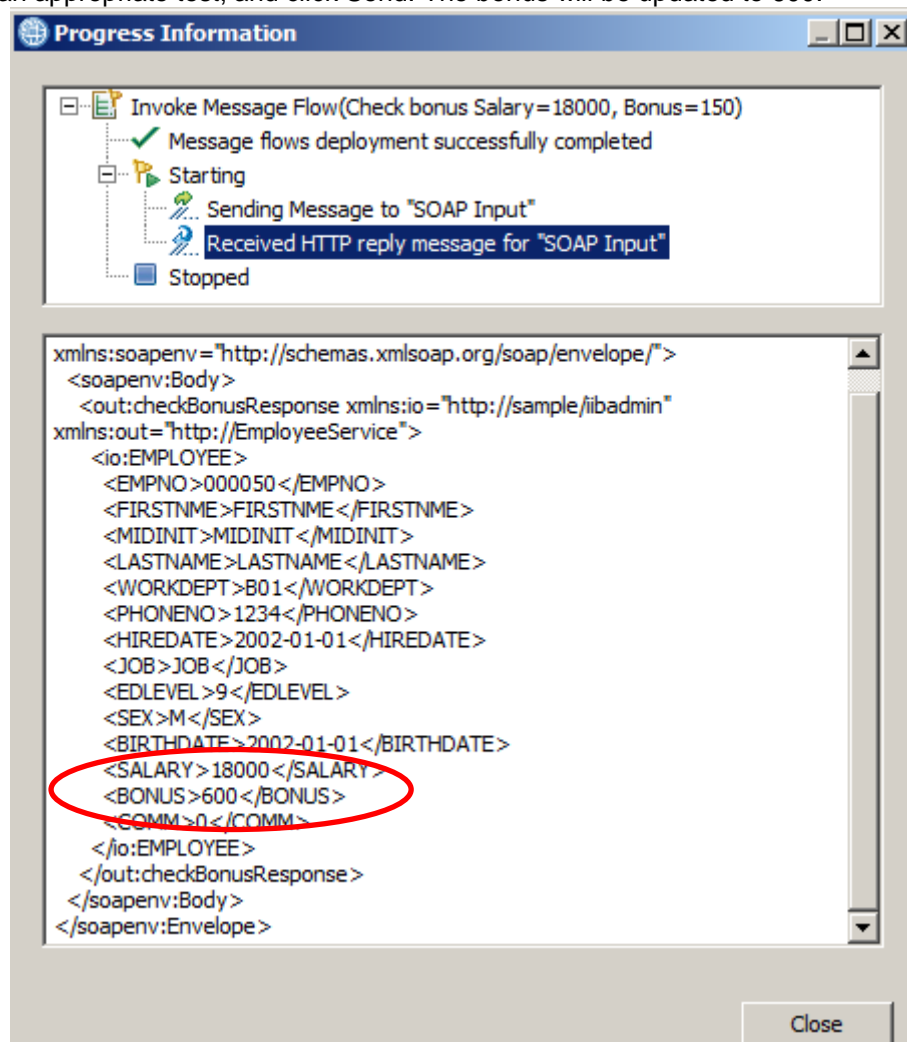
1. As before, open the Flow Exerciser with the red button.



Click the Send Message icon.



2. Select an appropriate test, and click Send. The bonus will be updated to 600.



**End of Decision Services lab.**