

# 7 Adding Business Rules

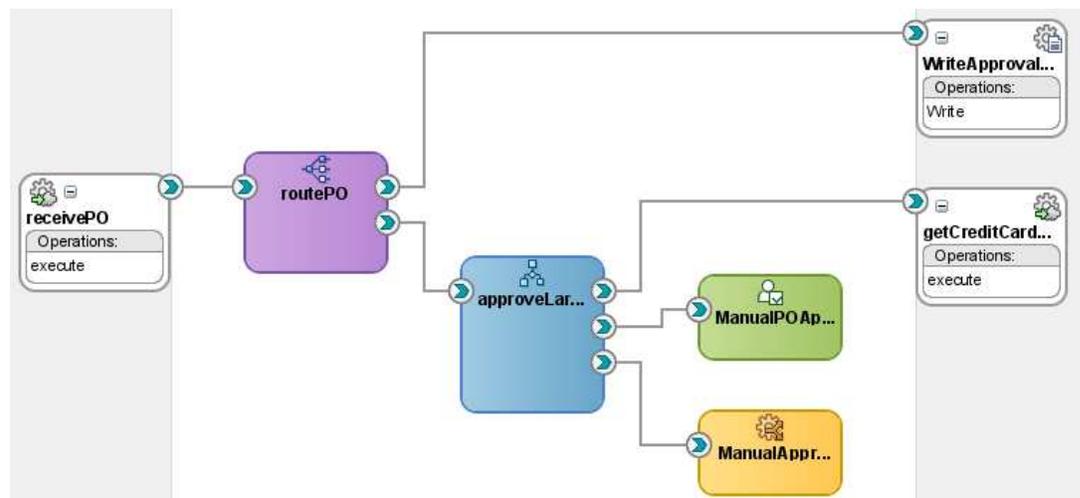
7.1	Introduction.....	1
7.2	Designing the flow .....	1
7.3	Adding a Business Rule to POProcessing .....	2
7.4	Deploying POProcessing composite.....	11
7.5	Running the application.....	12
7.5.1	Additional Test Cases .....	13
7.1	Operations and naming.....	14

## 7.1 Introduction

**Note:** The solution for this chapter can be found in `c:\po\solutions\ch7`.  
To run this solution, you must first complete Chapter 1 to set up the application.

In this chapter you will add a Business Rule to the composite to make the decision of whether the human task is required for manual approval or not. When you added the human task, it would be called for orders > \$1000. Now, the Business Rule will determine whether or not the human task is required.

When this chapter is completed, the composite looks like this.



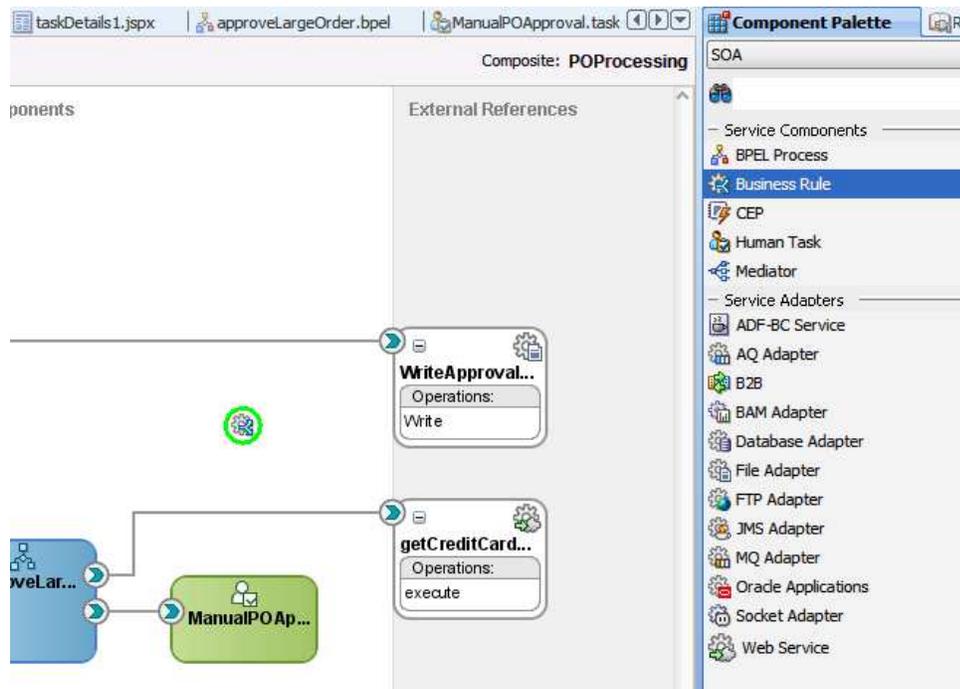
## 7.2 Designing the flow

The Business Rule uses the order total to determine whether or not manual approval is required. Orders over \$5000 require manual approval otherwise, the order is approved automatically. The BPEL process only sends orders to the Business Rule that have already had the credit card validated.

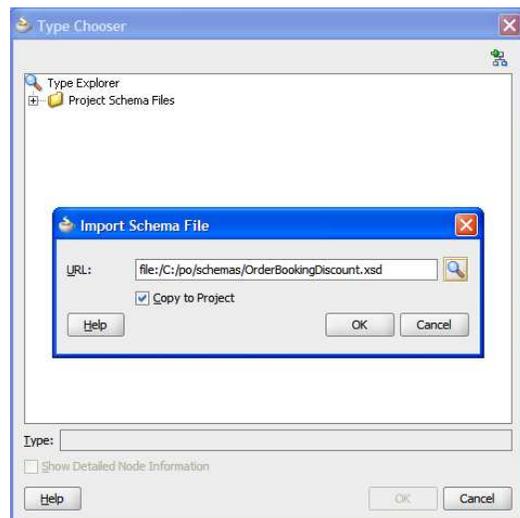
## 7.3 Adding a Business Rule to POProcessing

You first add the Business Rule component to the composite and then you invoke the rule from the BPEL process.

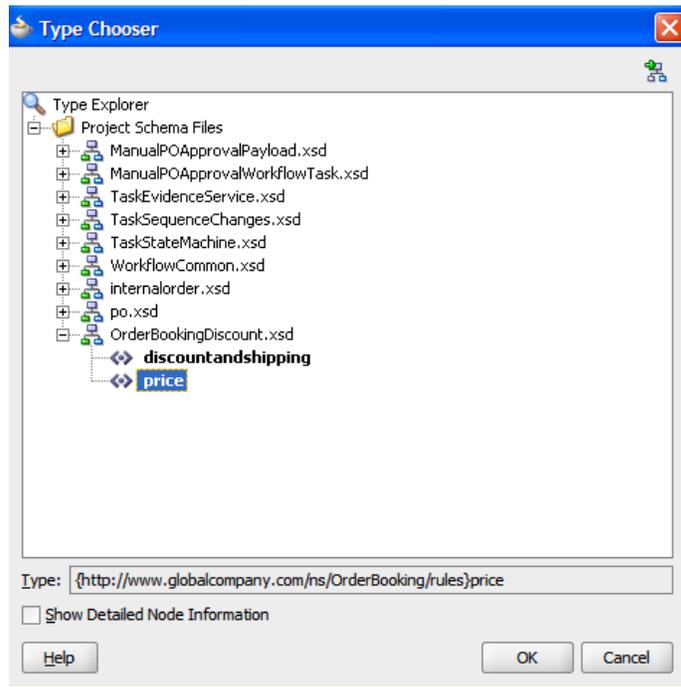
1. Add the Business Rule component to the composite by dragging it to the Components section.



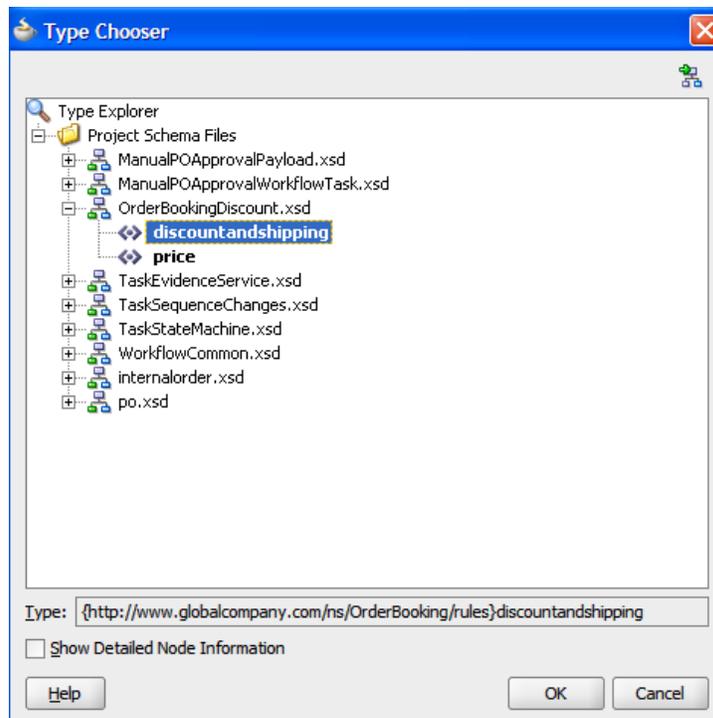
2. In the Create dialog enter *ManualApproval* for the Dictionary name.
3. Now set the input and output schema. Selecting **Input** from the green plus-sign drop down opens the Type chooser for choosing the schema. Select the top right button to import a schema and then browse to the schema location:  
[c:\po\schemas\OrderBookingDiscount.xsd](file://c:/po/schemas/OrderBookingDiscount.xsd)



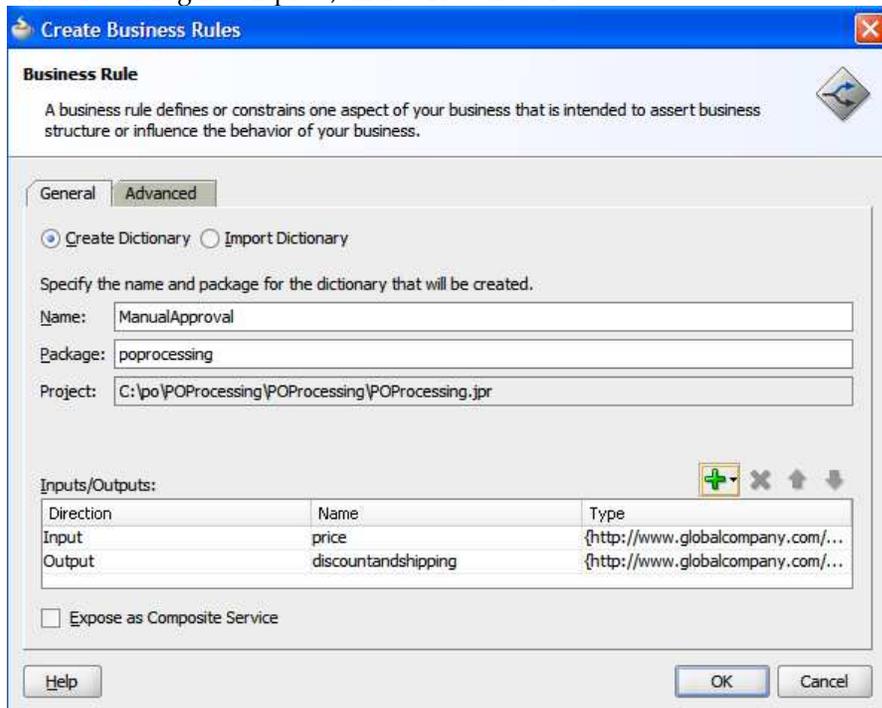
4. Select `price` for the input schema type.



5. Repeat the steps for the output schema. This time you do not need to import the schema type. Just select the type `discountandshipping`.



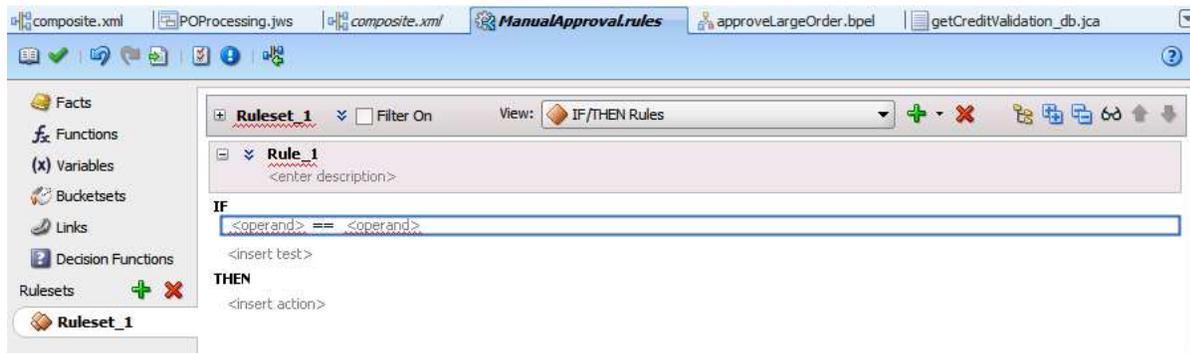
- Once the dialog is complete, click **OK** to close it.



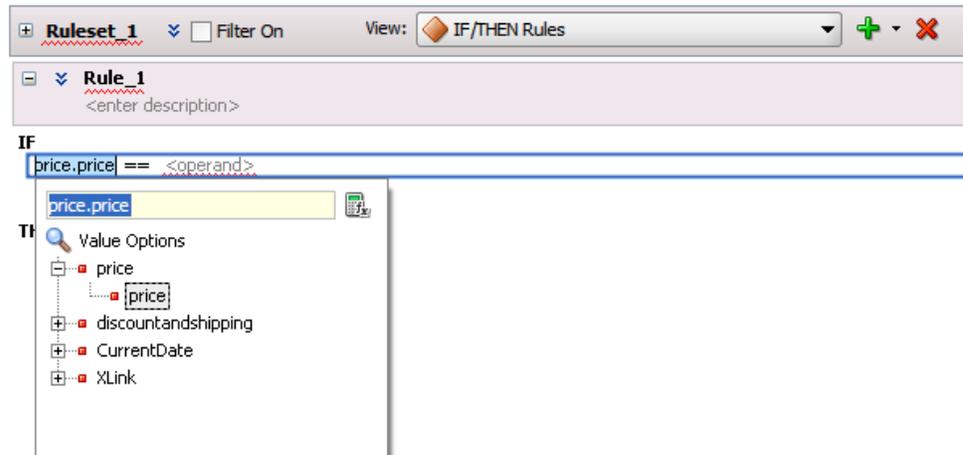
- Double-click the Rules component to open the rule editor. Select RuleSet\_1



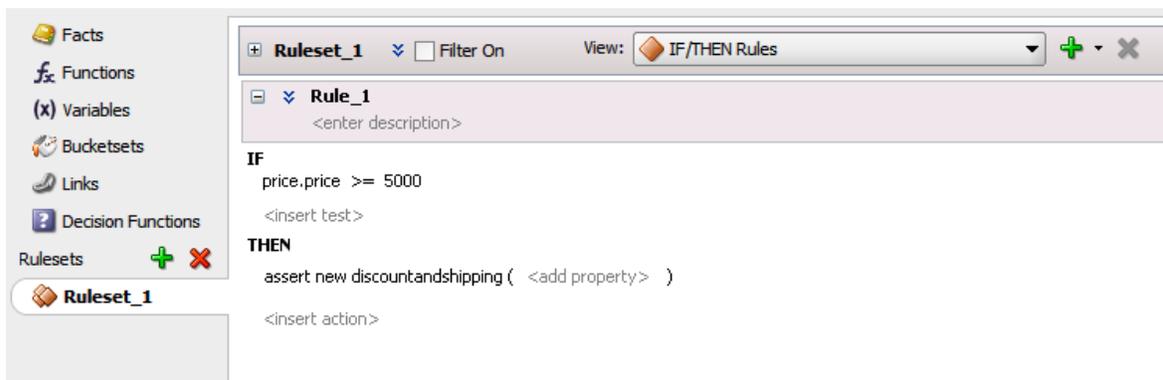
- Click the green plus sign to add a rule template.



9. Select **insert test** and then select **operand** in the IF statement and select the price.price object.



10. Complete the test for  $\geq 5000$ , pressing Enter to enter the value and add the THEN clause to create the return result.

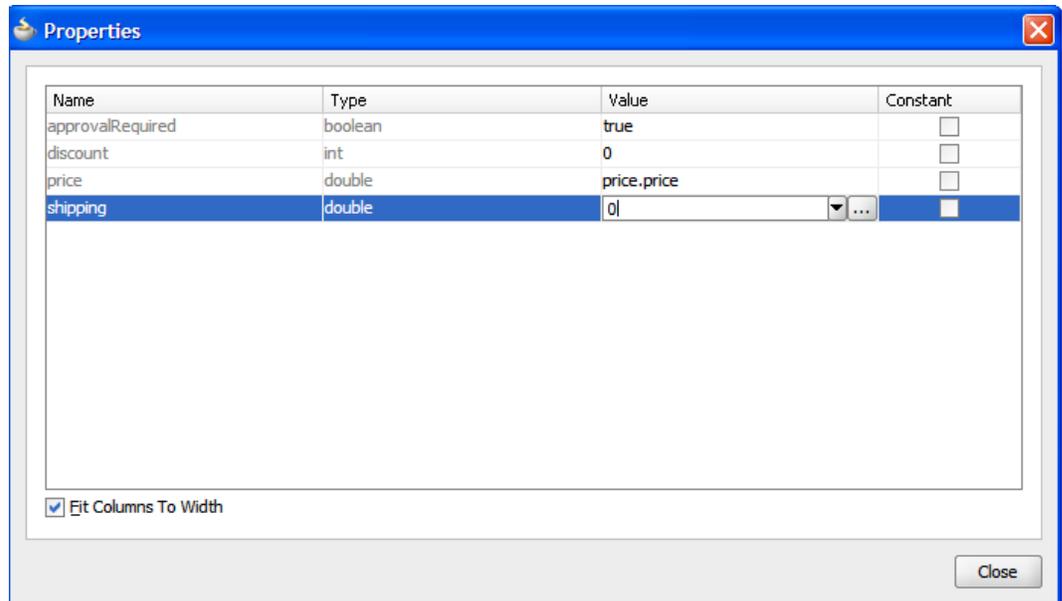


11. Select **<insert action>** and select **assert new**. Select **<target>** and select *discountandshipping*.
12. Select the **<add property>** box and a dialog comes up to set the property values. Set the values as shown. We are not using discount or shipping in this composite application so set those values to 0 (use Enter to enter the value). For orders \$5000 or more, set the approval required to true. You can also set the price value to the input price.  

```

approvalRequired = true
discount = 0
price = price.price
shipping = 0

```



13. Repeat the process to add a second rule for orders under \$5000. In this case, approval is not required.

Ruleset\_1 Filter On View: IF/THEN Rules

**Rule\_1**  
<enter description>

**IF**  
price.price >= 5000  
<insert test>

**THEN**  
assert new discountandshipping ( <add property> approvalRequired : true , discount : 0 , price : price.price , shipping : 0 )  
<insert action>

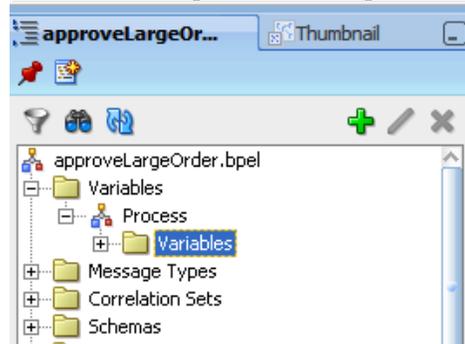
**Rule\_2**  
<enter description>

**IF**  
price.price < 5000  
<insert test>

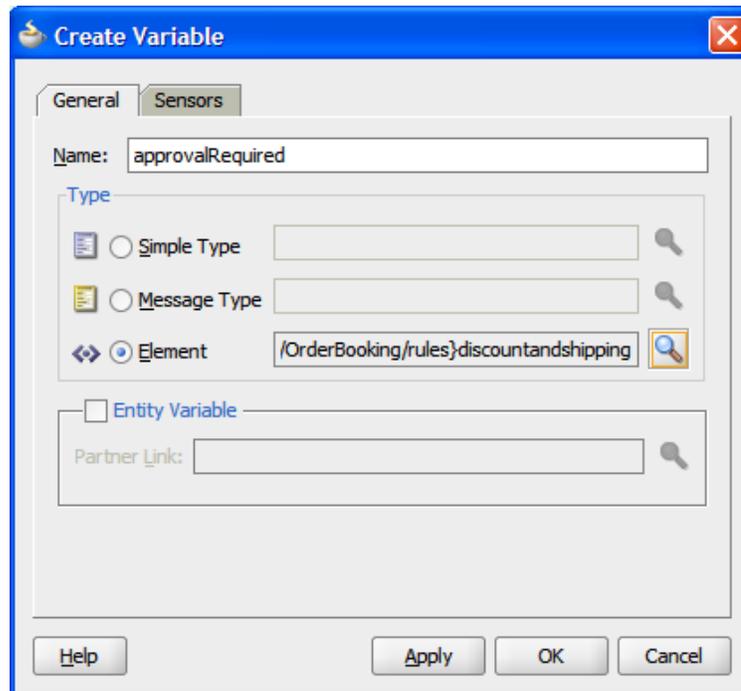
**THEN**  
assert new discountandshipping ( <add property> approvalRequired : false , discount : 0 , price : price.price , shipping : 0 )  
<insert action>

14. Save and close the Rules dialog.
15. Next, double-click the BPEL icon to open the BPEL process. You will be adding a Rule activity to just before the existing Human Task. Let's first create a variable to store the output from the RuleSet.

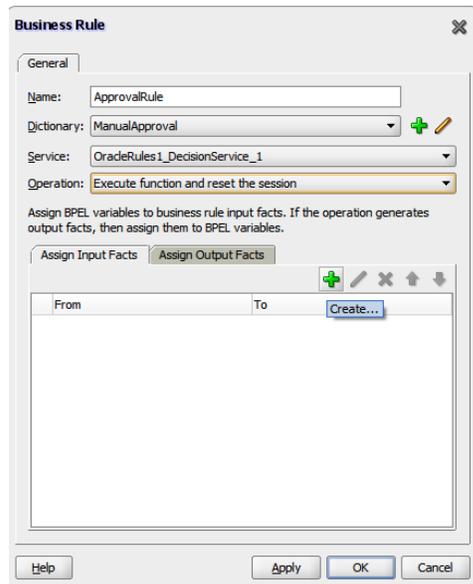
16. In the **Structure** palette, open **Variables/Process** and select the **Variables** node. If the Structure palette is not open, select it from the **View** menu.



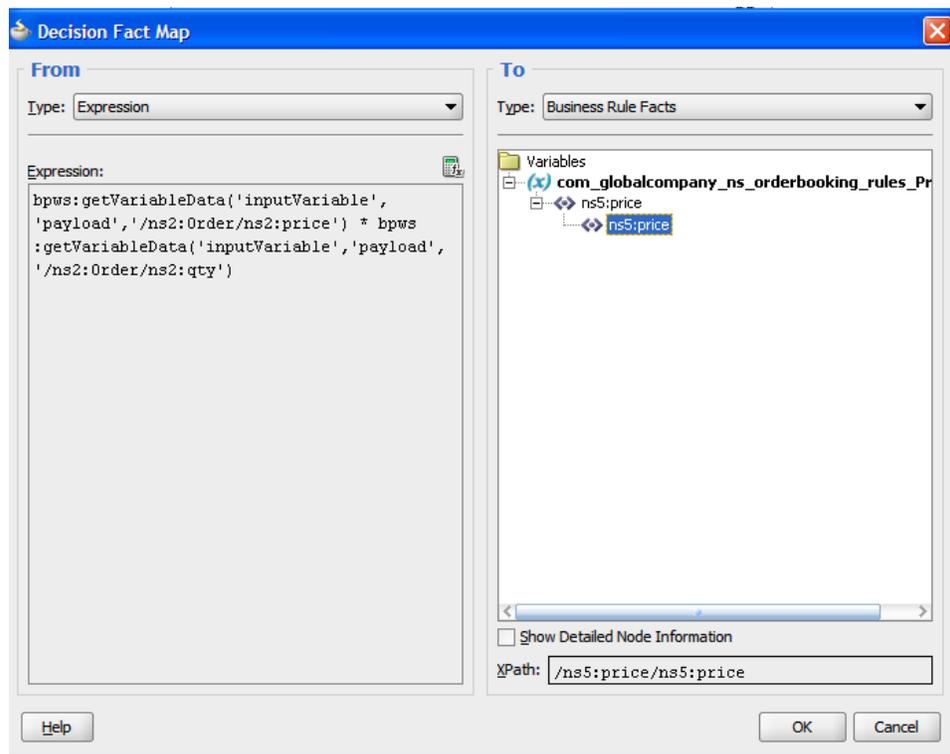
17. Select the green plus-sign to add a variable named *approvalRequired* of type *discountandshipping*.



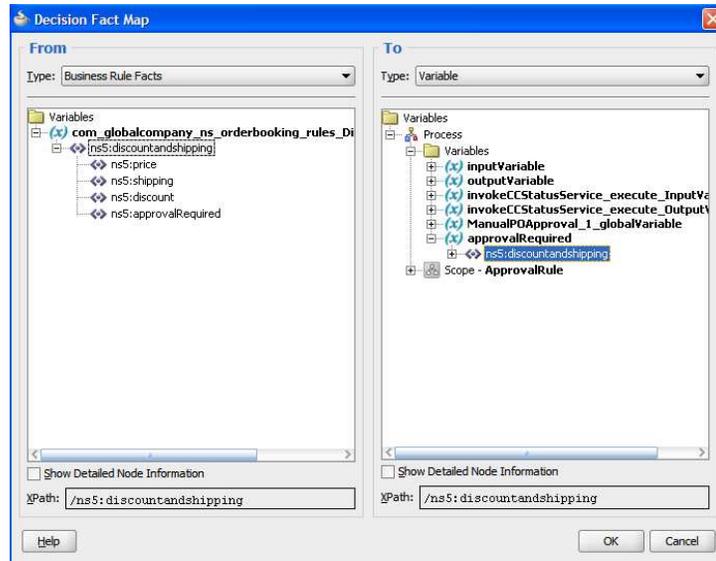
18. Now drag the Business Rule to the location just before the Human Task, within the case block of the switch activity.
19. The **Rule** dialog will open. Set the **Name** and select the **Business Rule Dictionary** you just created in the composite. Next complete the **Input Facts** and **Output Facts**.



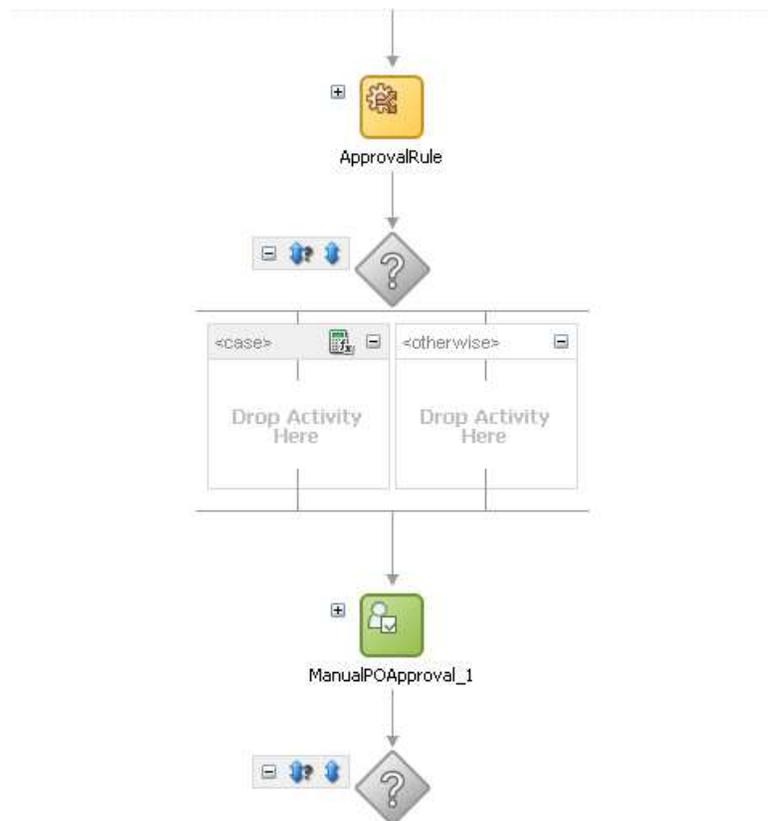
20. The input fact will be the price value. You create an expression using the expression builder to multiply the item price by the quantity.



21. After the RuleSet is evaluated, the output fact is returned. Set the value of the output fact to the variable you just created. Make sure to select the whole payload to copy all the values over (although we are only interested in the approvalRequired value).



22. Click **OK** to close the dialog.
23. Now, copy a **Switch** activity to below the **Rule** activity and expand the case blocks.



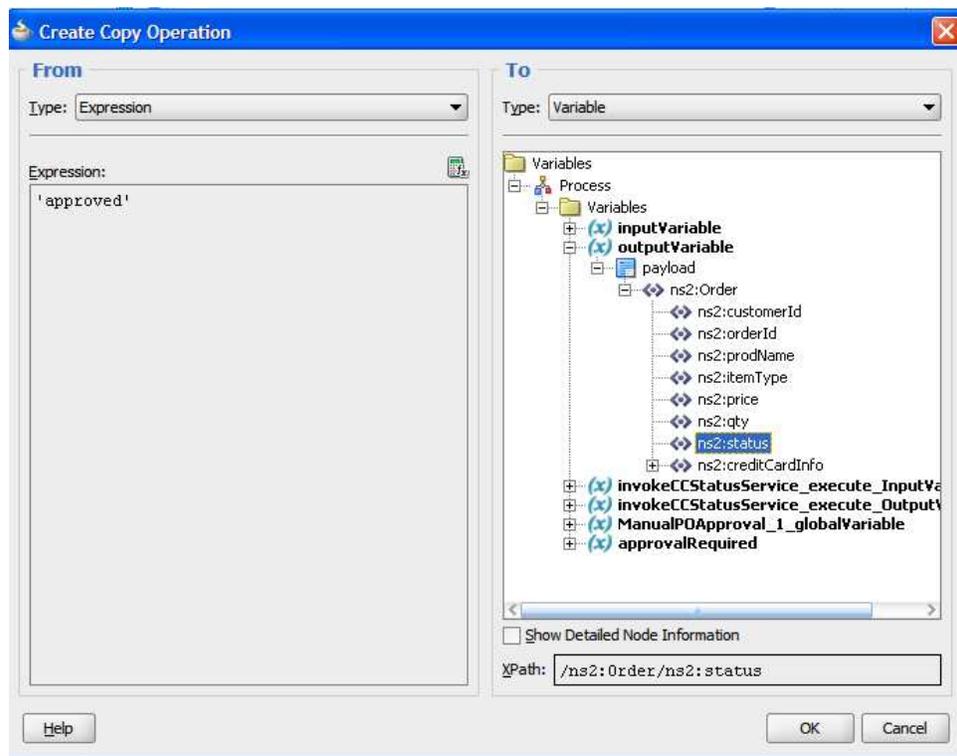
24. Select **Save All** before continuing.

You will be moving the human task and switch statement into the case block of this new **Switch** activity. Before you do that, drag a **Sequence** activity to keep the moved activities in the proper order.

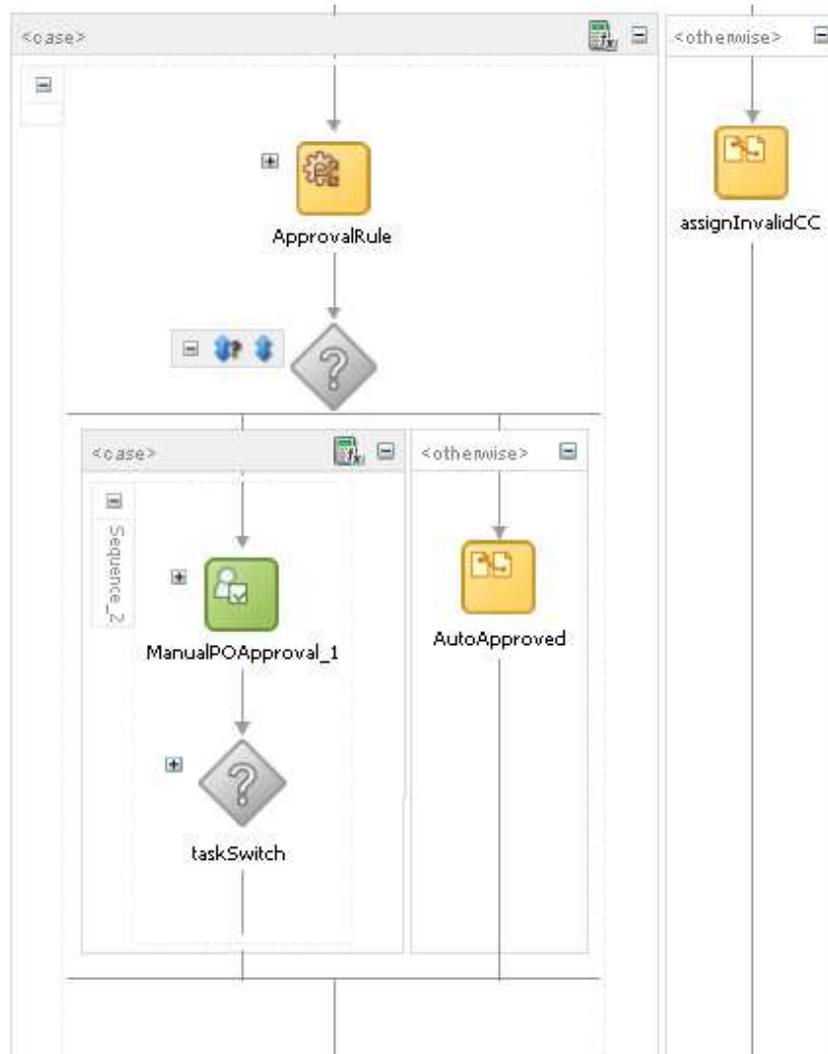
25. Drag a **Sequence** activity from the palette into the case block.
26. Move the **Human Task** into the **Sequence** block for this switch so that it only executes when the test case is true.
27. There is a **Switch** activity that follows **Human Task** for processing the **Human Task** results which sets the status to approved or rejected accordingly. Move this into your new Sequence block in the switch activity case statement as well.
28. Select **Save All** before continuing.
29. Use the **Expression Builder** to set the test case expression in the case block. You want to invoke the human task only when manual approval is required. This is the result from the Rule evaluation above:

```
bpws:getVariableData('approvalRequired', '/ns5:discountandshipping/ns5:approvalRequired') = 'true'
```

30. Next create an assign activity in the **Otherwise** block of your new **Switch**. Call this assign, *AutoApproved*. This assigns the status value approved for the case where the manual task is not required.



31. Your completed BPEL changes look like this.



Review the logic of the changes you made: When the credit card is valid, you ask the Rule to tell you if manual approval is required. If manual approval is required, call the Human Task service to take care of that and set the status based on the returned value. If it isn't required, auto approve the order in the otherwise block.

You have completed all changes and are ready to deploy. Continue with the next section.

## 7.4 Deploying POProcessing composite

32. In the same way as you did before, deploy your newly updated POProcessing. Using the project menu, select **Deploy**.
33. When you see the **Deployment Plan** dialog, enter a new version or select the **Overwrite Version** checkbox and click **OK**.

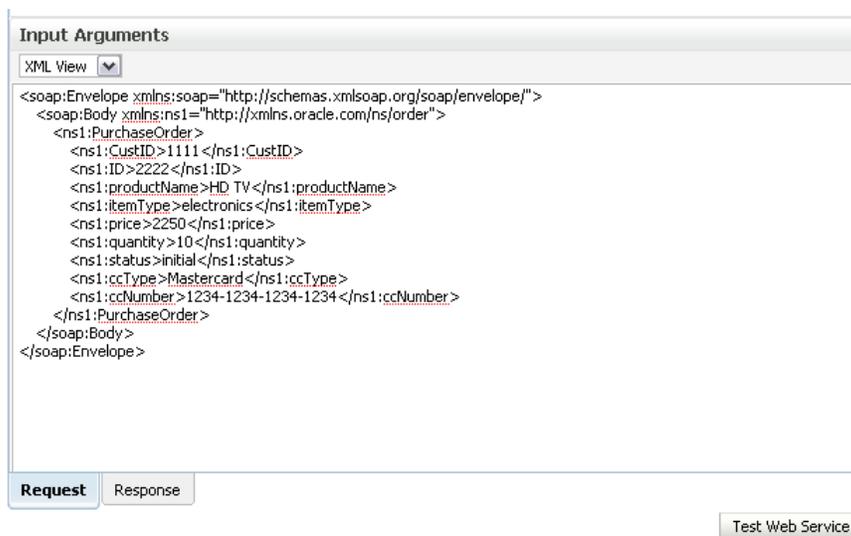
## 7.5 Running the application

34. Once the application is deployed, you are ready to try running it. Open the EM console at <http://localhost:7001/em>
35. Click on **POProcessing** and then the **Test** button to test your service.
36. Enter an extra large order ( $\geq 5000$ ) by doing one of the following:

- Type the values into in to the HTML form, or
- Click **XML View** so you can paste in the XML payload. This is the recommended way. Open the following file in a text editor:

```
c:\po\input\po-xtralarge-HDTVx10.txt
```

Copy the entire contents and paste them into the large text field in your browser, replacing the defaulted xml:



37. Click **Test Web Service**.
38. The **Response** screen won't have any response because this is a one-way invocation with no reply or callback.

There is also no file written to the output directory. That's because the application hasn't completed running yet. To see what's going on, you need to look at the instance audit trail.

39. Click **Launch Message Flow Trace** to see the details of the message flow for this instance of your composite. Alternatively, click the latest instance from the composite page. You'll notice the status for **approveLargeOrder** and **ManualPOApproval** are both still running. They are waiting for the human task to be processed.
40. Login to the worklist app at <http://localhost:8001/integration/worklistapp> (use the correct port for your installation) with user *weblogic/welcome1* and approve the task.

41. Return to the task flow in the EM console. If you still have it open you can just refresh the page. Notice that it is now complete.

#### Trace

Click a component instance to see its detailed audit trail.

Show Instance IDs

Instance	Type	State	Time	Composite Instance
receivePO	Service	Completed	Mar 3, 2009 10:06:22 PM	POProcessing of 10003
routePO	Mediator Component	Completed	Mar 3, 2009 10:06:49 PM	POProcessing of 10003
writeApprovalResults	Reference	Completed	Mar 3, 2009 10:06:49 PM	POProcessing of 10003
approveLargeOrder	BPEL Component	Completed	Mar 3, 2009 10:06:49 PM	POProcessing of 10003
getCreditCardStatus	Reference	Completed	Mar 3, 2009 10:06:23 PM	POProcessing of 10003
getStatusByCC	Service	Completed	Mar 3, 2009 10:06:23 PM	validationForCC of 10004
RouteRequest	Mediator Component	Completed	Mar 3, 2009 10:06:23 PM	validationForCC of 10004
getCreditValidation	Reference	Completed	Mar 3, 2009 10:06:23 PM	validationForCC of 10004
ManualApproval	Decision Service Comp	Completed	Mar 3, 2009 10:06:28 PM	POProcessing of 10003
ManualPOApproval	Human Workflow Comp	Completed	Mar 3, 2009 10:06:48 PM	POProcessing of 10003

## 7.5.1 Additional Test Cases

There are now four test cases for your application depending on the input data value of the total price:

- under \$1000 (price x quantity) – auto approval without BPEL component
- invalid credit card – auto reject using BPEL but no Rules or Task components
- 1000 to under 5000 - auto approval using BPEL and Rules but no Task
- 5000 and over – manual approval using BPEL, Rules and Task components

The first case has been provided for you above. An invalid credit card number is: [4321-4321-4321-4321](#). When you get to the manual approval case, follow these instructions to see the human task.

42. Open the EM console to test the application using a very large input value in the same way as you did in the previously for a small order. Create a new order that's over \$5,000. You can use the provided input data file [po-xtralarge-HDTVx10.xml](#)
43. As usual, the **Response** page won't show any results as this is a one-way invocation with no reply or callback. Also notice, however, that a new order file was not created in `c:\temp`. That's because the application hasn't completed running yet. To see what's going on, you need to look at the instance audit trail.
44. Click **Launch Message Flow Trace** to see the details of the message flow for this instance of your composite. Alternatively, click the latest instance from the composite page.
45. Click on the **approveLargeOrder** BPEL component.
46. Scroll to the bottom and notice the BPEL process is incomplete, blocked on the Human Task until it is approved.
47. Now it is time to "change hats" and log on to the worklist application as a customer service representative to approve the task. Using a browser, navigate to <http://localhost:8001/integration/worklistapp>

48. Logon using the following credentials:
  - **Username:** *weblogic*
  - **Password:** *welcome1* (depending on your installation)
49. The worklist application is an out-of-the-box application where users can view and manage their tasks.

Click on the most recent task to highlight it. The task shows in the bottom pane.
50. From the **Actions** menu at the top-right, select **Approve**. This will approve the task and notify the BPEL process it can continue.
51. Return to the BPEL process flow. If you still have it open you can just refresh the page.

Notice that it is now complete.
52. You will also have a new order file in `c:\temp`.

Continue with different input data values and try all four test cases.

## 7.1 Operations and naming

This section gives you all of the operations and names for objects created in this chapter. Experienced users can use this for creating the objects in this chapter quickly. Any questions on details for a particular operation listed here can be found in the preceding sections. The information is divided by the sections in this document.

Adding a Business Rule to POProcessing

- **Composite Business Rule:** ManualApproval
  - **Input:** OrderBookingDiscount.xsd > price
  - **Output:** OrderBookingDiscount.xsd > discountandshipping
- **RuleSet\_1: Rule 1:** If price >= 5000 then assert new discountandshipping with approvalRequired = true.
- **RuleSet\_1: Rule 2:** If price < 5000 then assert new discountandshipping with approvalRequired = false.
- **BPEL variable:** approvalRequired **of type** discountandshipping
- **Business Rule just before the Human Task**
- **BusinessRule:** ApprovalRule, ManualApproval
  - **Input fact:** price\*quantity to price
  - **Output fact:** discountandshipping to approvalRequired > discountandshipping
- **Switch below Rule**

- **Case expression:**  
`bpws:getVariableData('approvalRequired','/ns5:discountandshipping/ns5:approvalRequired') = 'true'`
- **Case:** add Sequence, move Human Task and human task switch
- **Otherwise: Assign:** AutoApproved
  - Copy From: 'approved'
  - Copy To: outputvariable > status

The application is completed. Continue with Section 7.4 above to deploy and test your application.