
Using Agile Event Framework and Oracle BPEL to Enhance PLM Workflow

Tutorial

By Jun Gao, Agile PLM Development
October 2009

CONTENTS

Chapter 1

- Overview 4**
- About this Tutorial.....4
- Objectives and Scope4
- Software Used4
- Getting Started 5**
- Preparing the Required Environment.....5
- Extending PLM Workflow Functionalities 6**
- Creating the BPEL Process.....6
- Creating Agile Partner Links.....9
- Defining the Input Variables 11
- Creating Partner Links and Defining Operations..... 13
- Deploying and Invoking the BPEL Process in Agile PLM..... 30**
- Deploying the BPEL Process 30
- Configuring Agile PLM and Invoking the BPEL Process..... 31



Chapter 1

Overview

This chapter includes the following:

- About this Tutorial..... 4
- Objectives and Scope..... 4
- Software Used 4

About this Tutorial

This tutorial describes how to use the Agile Event Framework and Oracle BPEL (Business Process Execution Language) to extend the capabilities of Agile workflow.

Objectives and Scope

BPEL-based business processes exchange data with third party applications and in turn invoke Agile PLM Web service APIs to manipulate objects in Agile PLM.

In this tutorial you will use the Agile 9.3 Event framework to trigger BPEL-based business process to modify or impact workflow processes, or send notifications.

Software Used

Software used in this tutorial are:

- Agile PLM 9.3
- Oracle SOA Suite 10.1.3.4 (Avaialable at [Oracle Technology Network \(OTN\) Web site](http://www.oracle.com/technology/documentation/agile.html) <http://www.oracle.com/technology/documentation/agile.html>)
- Oracle JDeveloper Studio Edition 11g (Avaialable at [Oracle Technology Network \(OTN\) Web site](http://www.oracle.com/technology/documentation/agile.html) <http://www.oracle.com/technology/documentation/agile.html>)

Chapter 2

Getting Started

This chapter includes the following:

- Preparing the Required Environment 5

Preparing the Required Environment

Prepare the environment by completing the following:

- Make sure the Oracle SOA Suite is running.
- Start JDeveloper.
- Verify that the Agile Web services are accessible at: <http://<your-agile-server>/CoreService/services>.

Extending PLM Workflow Functionalities

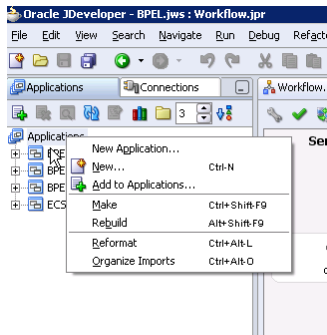
This chapter includes the following:

- Creating the BPEL Process 6
- Creating the Agile Partner Links 9
- Defining the Input Variables 10
- Creating Partner Links and Defining Operations 12

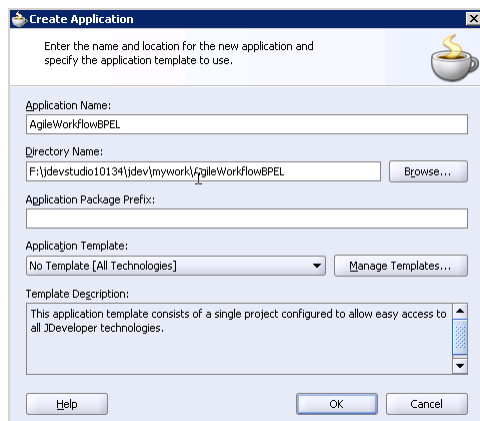
Creating the BPEL Process

Create the BPEL workspace and project using the following procedure.

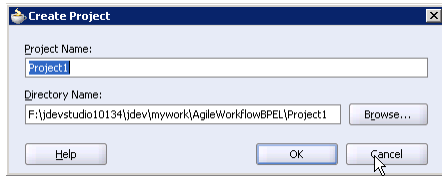
1. In the Applications Navigator window, right-click on Applications and select New Application....



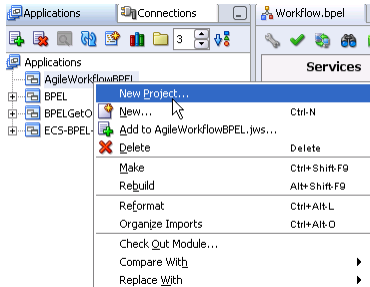
2. In application name field, type AgileWorkflowBPEL and accept all default settings (When the Directory Name does not look as shown below, ignore the message and accept the defaults).
3. Click OK.



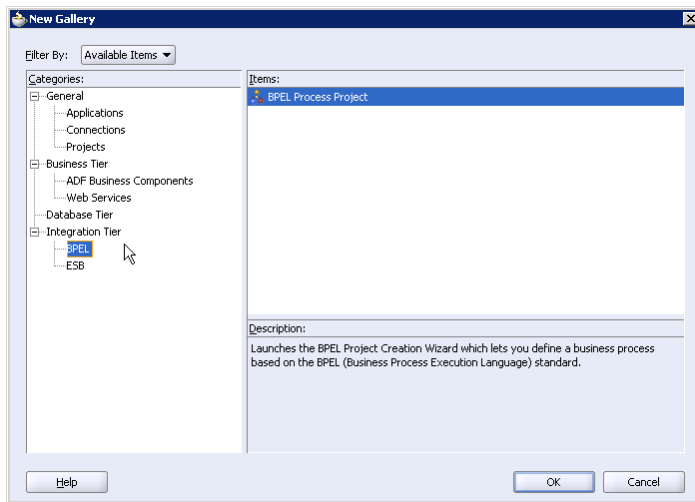
- Click Cancel.



- Right click the new AgileWorkflowBPEL application, and select New Project.



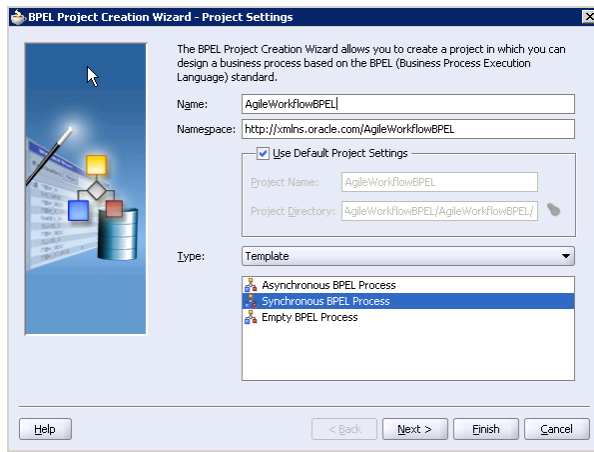
- In Categories list, select BPEL and in Items window, select BPEL Process Project.
- Click OK.



The BPEL Project Creation Wizard appears.

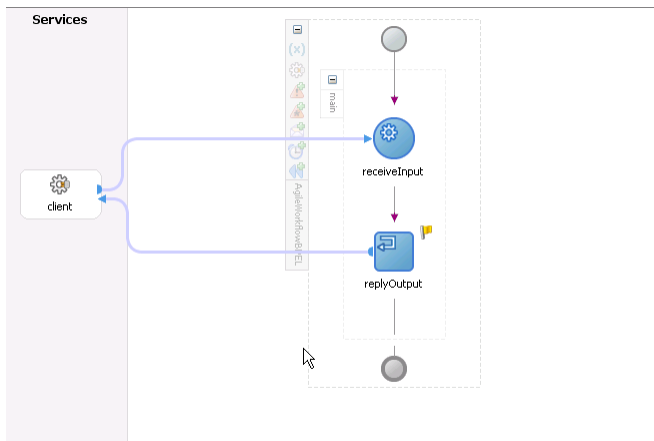
- In Project Settings window's Name field, type AgileWorkflowBPEL for process name.

9. Select Synchronous BPEL Process and then click Next.

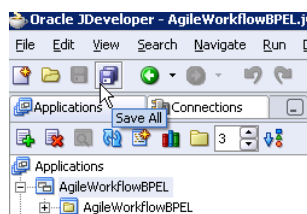


10. Click Finish.

The BPEL process diagram shown below, appears.



11. Click Save All to save your work.

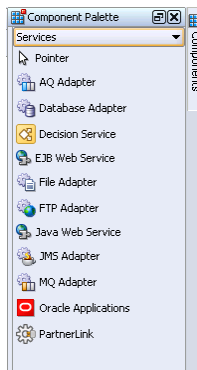


Chapter 4

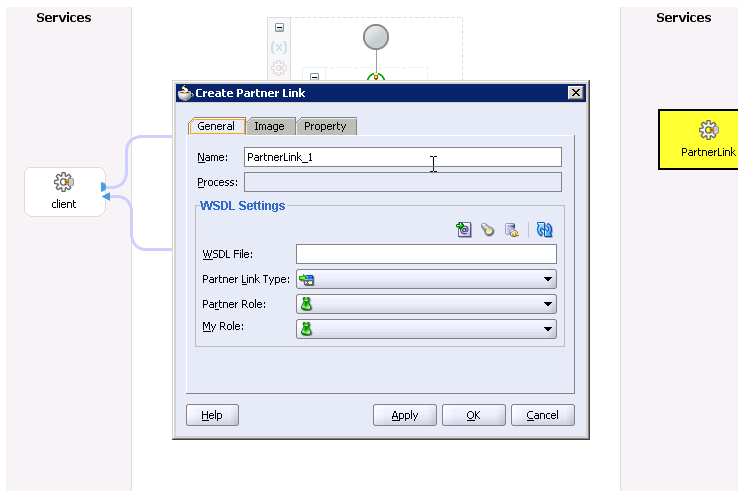
Creating Agile Partner Links

To create the Agile Partner Links, do as follows:

1. In upper right corner of JDeveloper, select Component Palette () and Services in the drop down menu.



2. Drag and drop a PartnerLink activity to the right side of the designer window anywhere beneath Services.



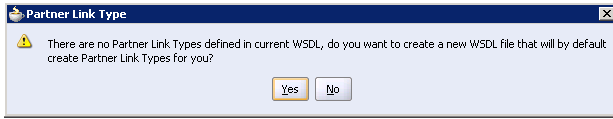
3. In Create Partner Link window, type the Agile WSDL URL in the WSDL File field and press ENTER.

The Oracle BPELProcess Manager will load WSDL from the Agile URL:

<http://93demo/CoreService/services/Collaboration?wsdl>

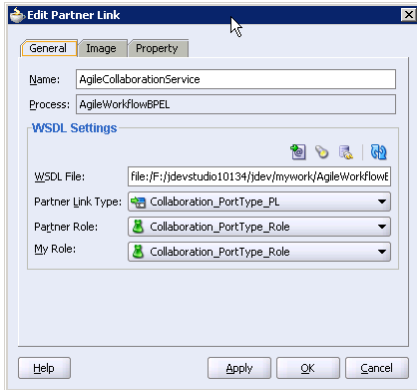
Note Replace 93demo in the above URL with the location of your Agile instance.

4. Click Yes.



The Name and Partner Link Type fields are automatically populated.

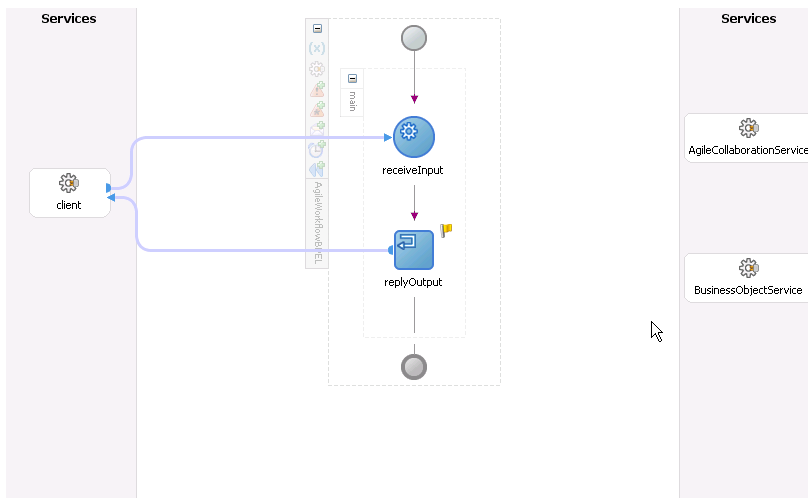
5. Select Partner Role as shown below.



6. Click OK.
7. Repeat steps 1 through 6 to create the AgileBusinessObject partner link.

Note In step 3, you must use this WSDL URL:
<http://93demo/CoreService/services/BusinessObject?wsdl>.

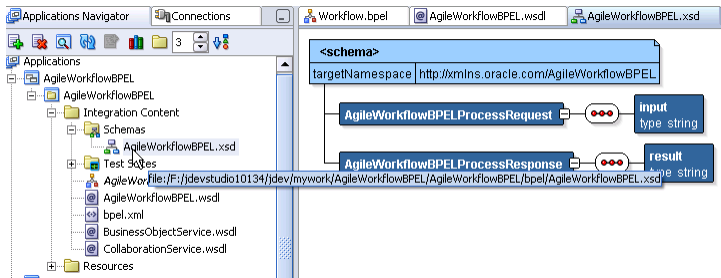
The BPEL designer shown below will appear.



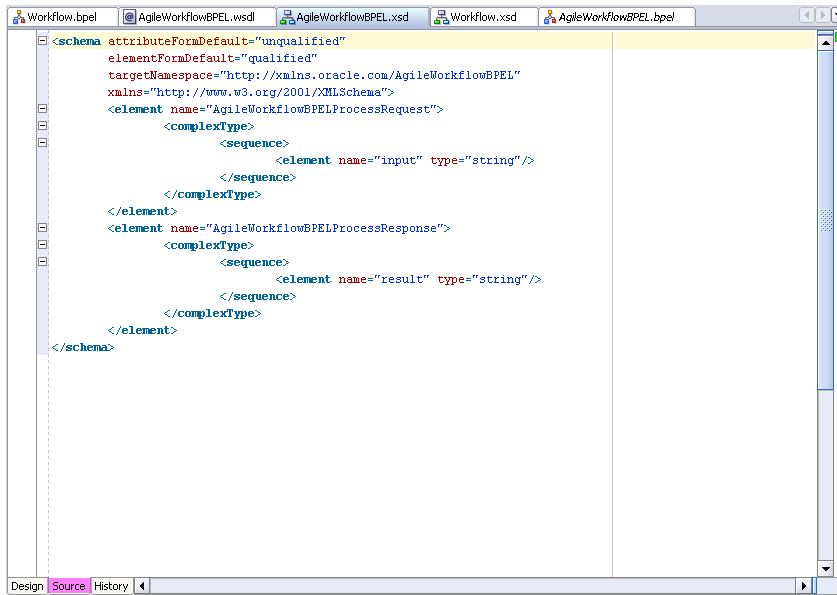
Defining the Input Variables

Define the input variables as follows:

1. Open AgileWorkflowBPEL.xsd under the AgileWorkflowBPEL project->Integration Content->Schemas in the left side panel



2. Switch to source view from design view by click the design tab

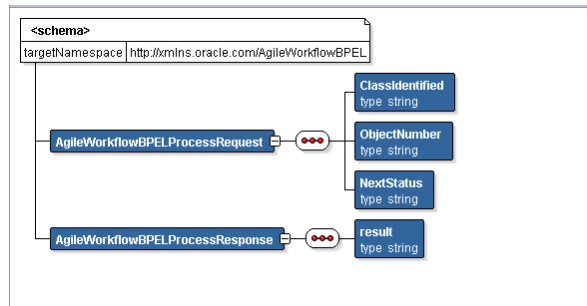


3. Replace the element `<element name="input" type="string"/>` with the following three elements
`<element name="ClassIdentified" type="string"/>`
`<element name="ObjectNumber" type="string"/>`

```
<element name="NextStatus" type="string"/>
```

```
<schema attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://xmlns.oracle.com/AgileWorkflowBPEL"
  xmlns="http://www.w3.org/2001/XMLSchema">
  <element name="AgileWorkflowBPELProcessRequest">
    <complexType>
      <sequence>
        <element name="ClassIdentified" type="string"/>
        <element name="ObjectNumber" type="string"/>
        <element name="NextStatus" type="string"/>
      </sequence>
    </complexType>
  </element>
  <element name="AgileWorkflowBPELProcessResponse">
    <complexType>
      <sequence>
        <element name="result" type="string"/>
      </sequence>
    </complexType>
  </element>
</schema>
```

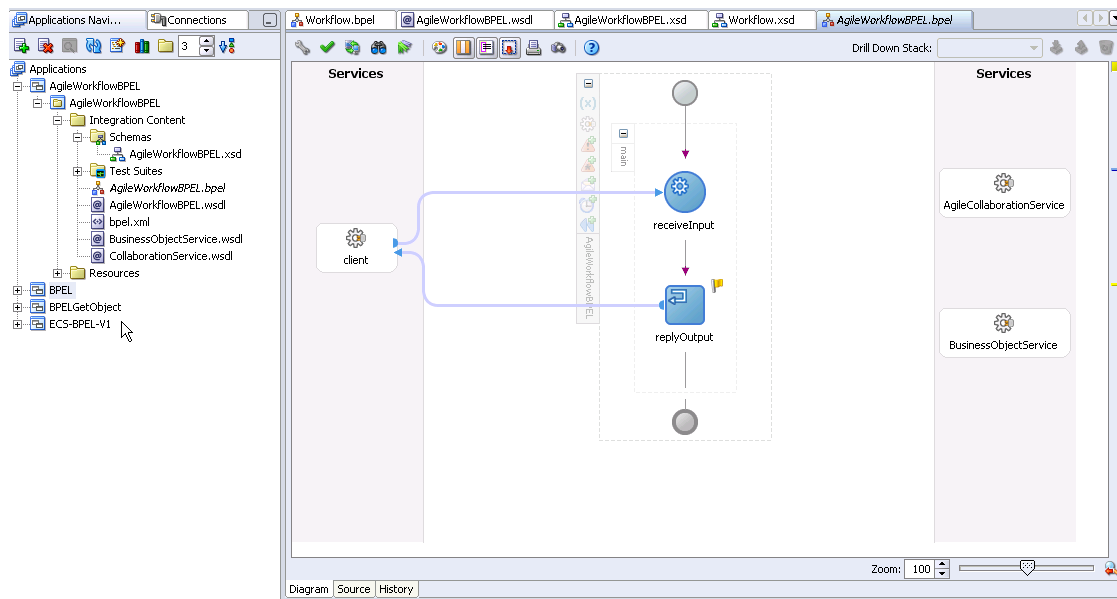
4. Click the Design tab to switch to Design view.



Creating Partner Links and Defining Operations

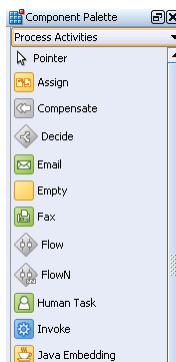
To create Partner Links and define Operations, do as follows:

1. In the left side panel of the BPEL design diagram, select AgileWorkflowBPEL.bpel under AgileWorkflowBPEL project >Integration Content >Schemas.

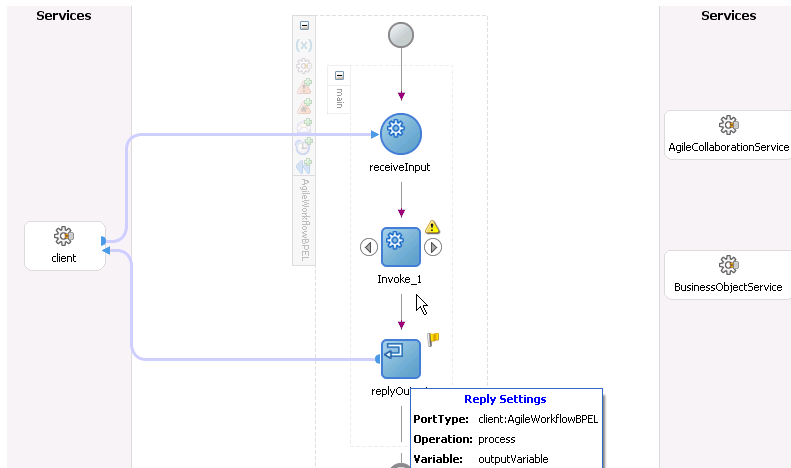


2. In JDeveloper, select Component Palette and Process Activities from the drop down list.

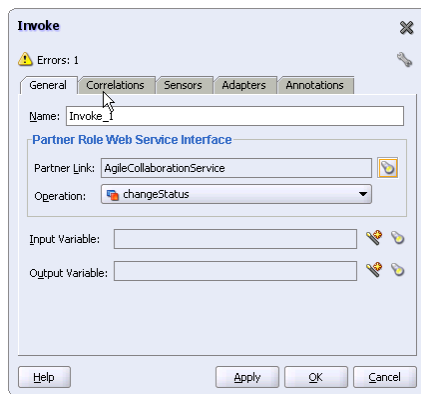
The Process Activities displays the BPEL activities that you can drag and drop into the process designer window.



3. Drag and drop an Invoke from the Component Palette section onto the canvas immediately after the receiveInput activity.

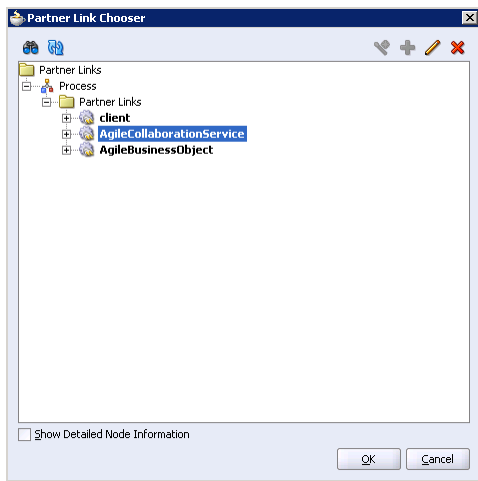


4. Double click the Invoke icon to open the Invoke window

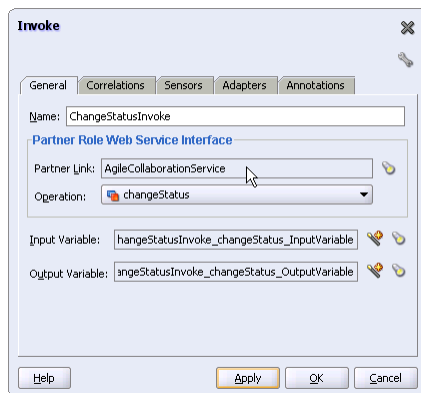


5. Set Invoke Name to UpdateObjectInvoke
6. Click Flashlight.

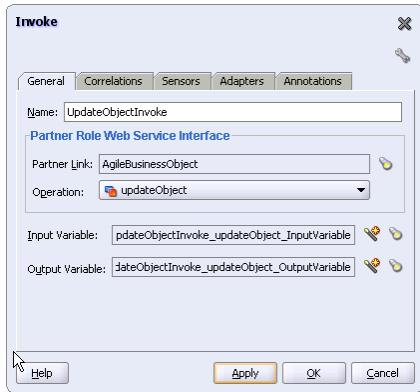
7. Select the AgileCollaborationService partner link and click OK.



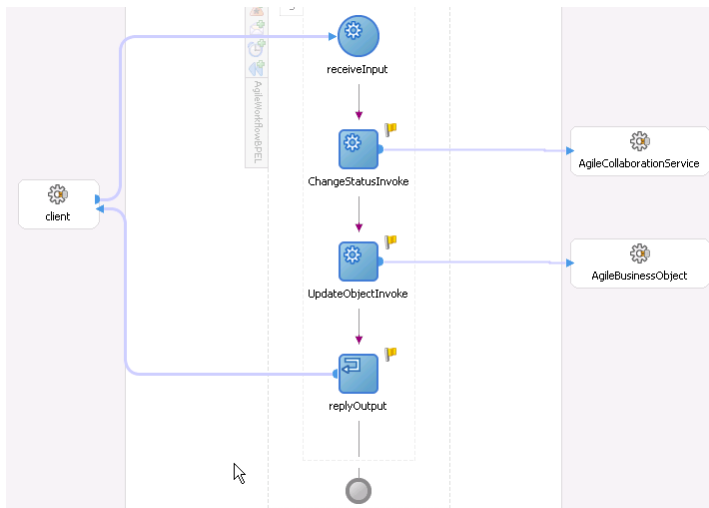
8. Set operation as changeStatus.
9. Click on the 'Auto-create variable' icon next to the Input Variable text field to create a global variable for this activity. Accept defaults. Click Apply and then OK.
10. Click Auto-create variable next to the Output Variable text field to create a global variable for this activity.
11. Accept defaults. Click Apply > OK.
12. Make sure your window looks as shown below and then click OK.



- Repeat steps 1 - 13 to create UpdateObjectInvoke.

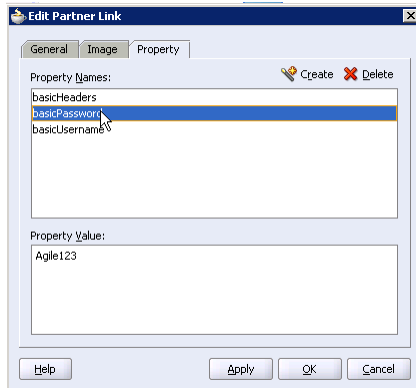
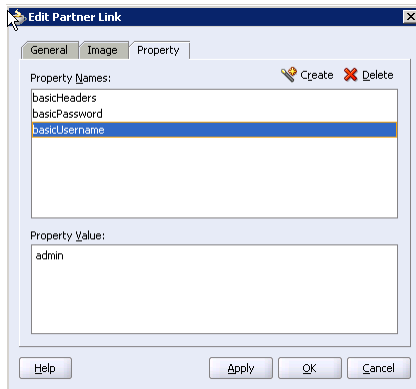
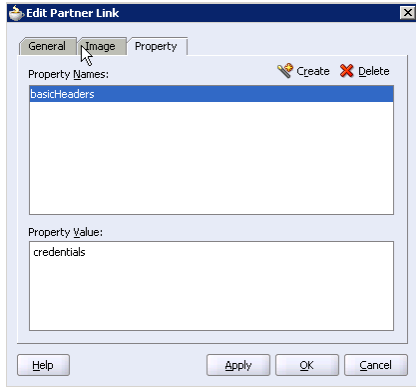


The BPEL design area shown below will appear.



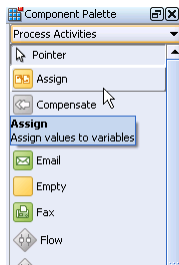
- Set the credentials for the above two partner links.
- Double Click the partner link "AgileCollaborationServer" icon, and then go to the property tab to create two properties.

Note You must validate the values used for username and password against the Agile PLM system.

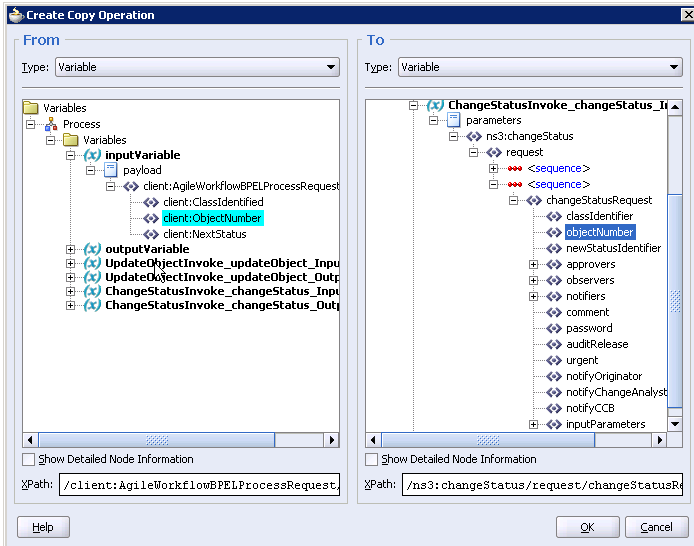
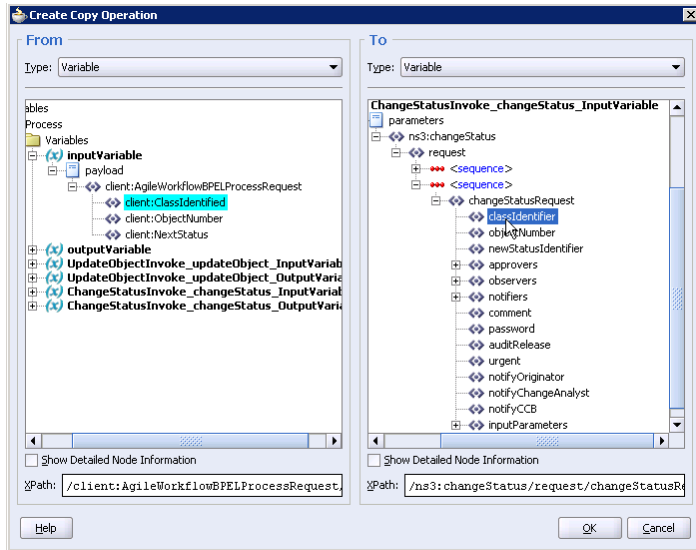


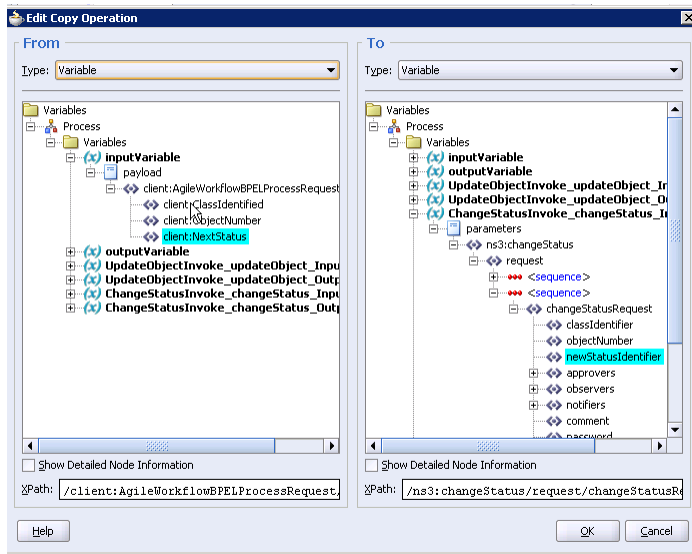
16. Repeat above step to create the AgileBusinessObject Partner link.

17. Drag and drop an Assign activity from the Component Palette section to receiveInput and ChangeStatusInvoke activities.

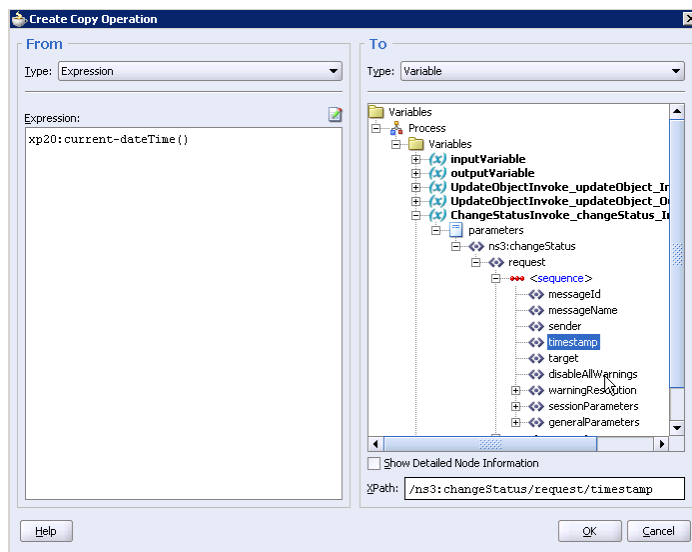


18. Double click on the recently added Assign icon.
19. Click the General tab and type CopyInput in the Name field.
20. Click Apply.
21. Click the Copy Operation tab.
22. Click Create and then select Copy Operation from the drop down list.
23. Create two Copy Operations as shown below. Be sure to click OK after every copy operation to continue.

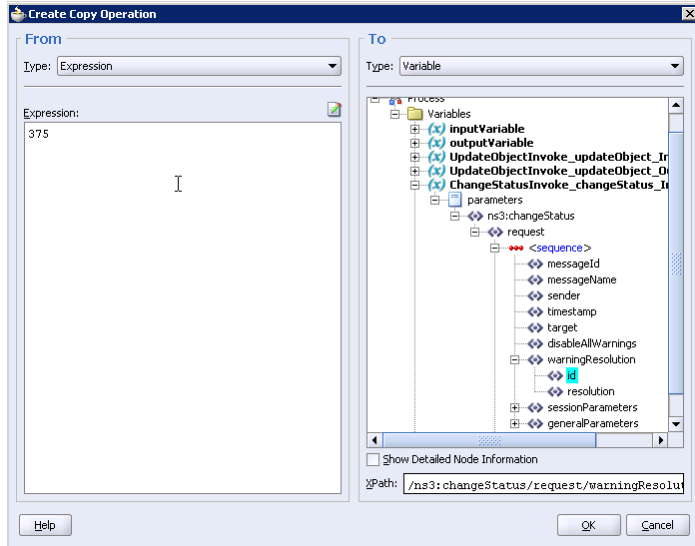




24. Click Create and then select Copy Operation from the drop down list.
25. In From column, select Expression from the Type drop down list and type xp20:current-dateTime().

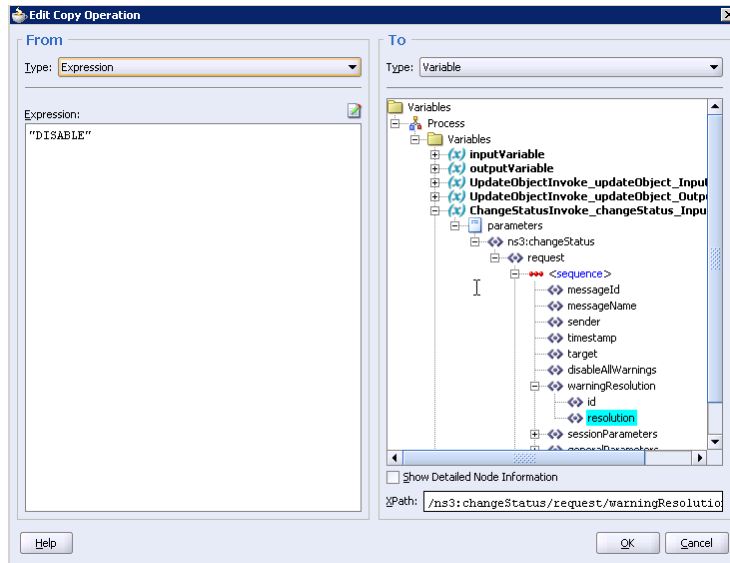


26. Click Create and then select Copy Operation from the drop down list.
27. Select Expression from the Type drop down list in the From column and type 375.



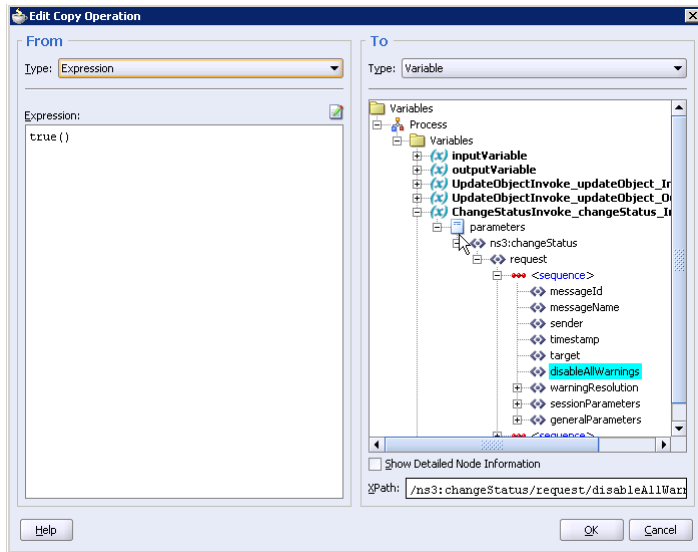
28. Click Create and then select Copy Operation from the drop down list as shown above.

29. In From column, select Expression from the Type drop down list.,and type DISABLE.

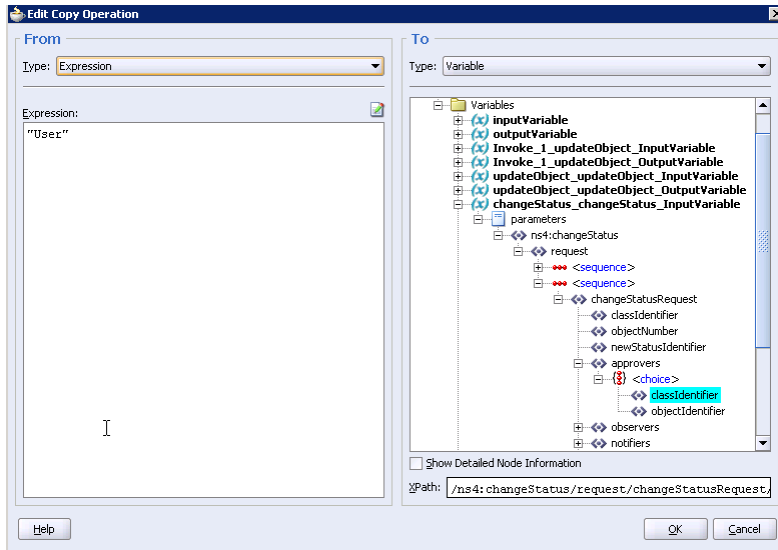


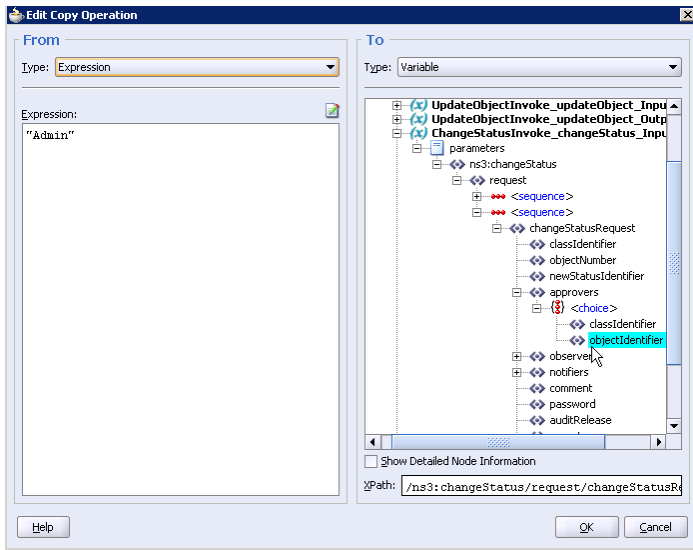
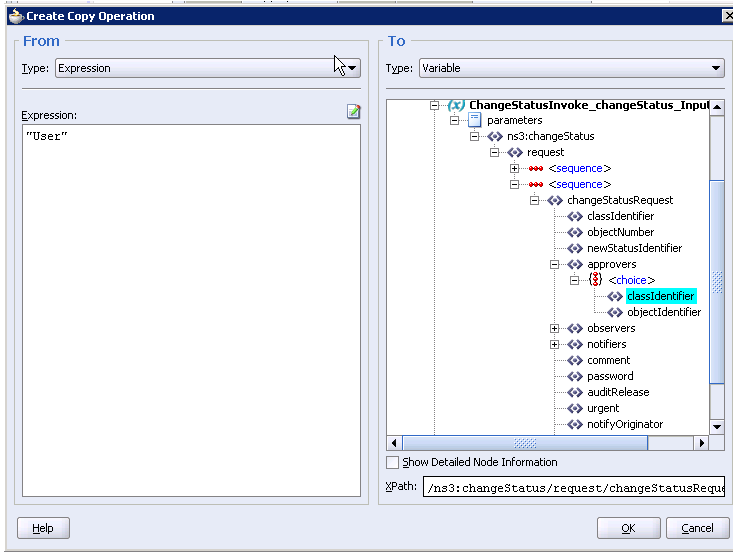
30. Click Create and then select Copy Operation from the drop down list.

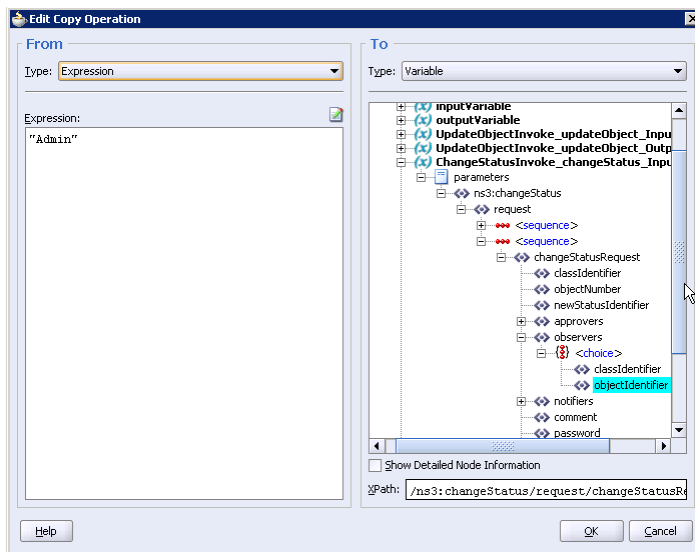
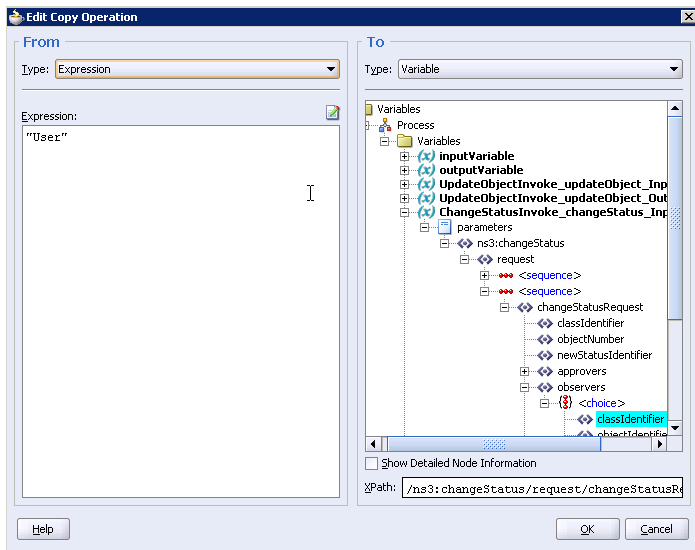
31. Select Expression from the Type drop down list in From column and type true().

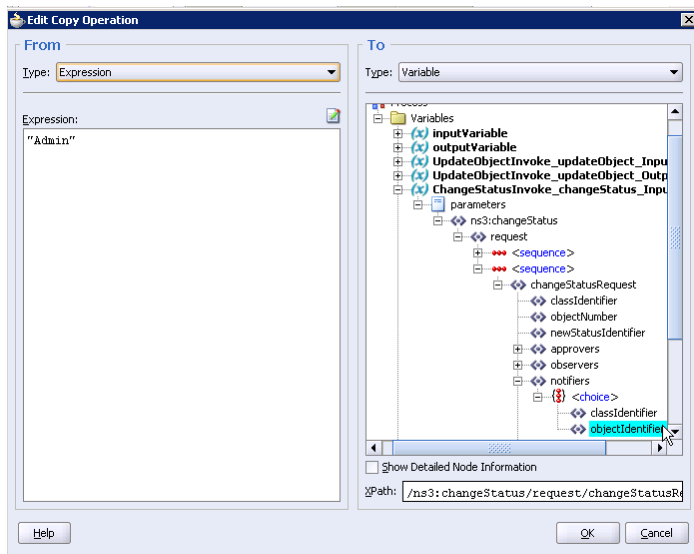
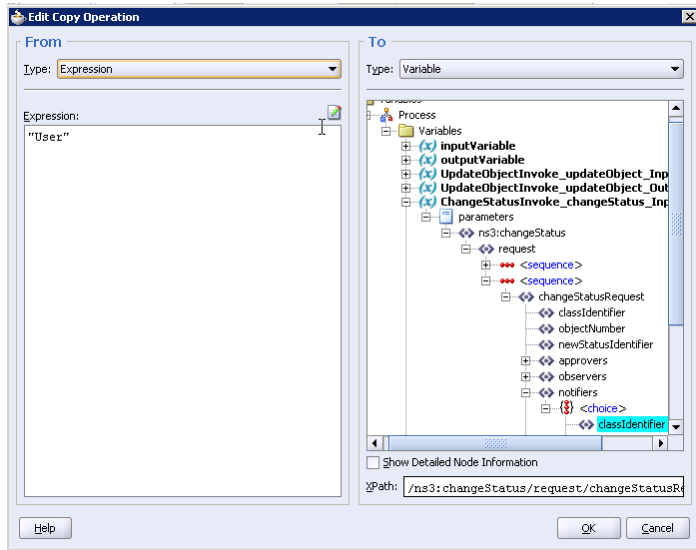


32. Create a few more copies of Operations as shown below and replace admin with a valid user ID in your system.



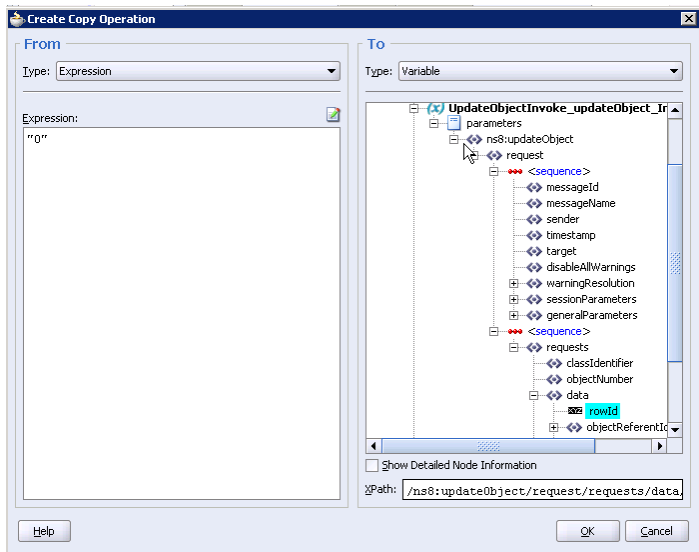
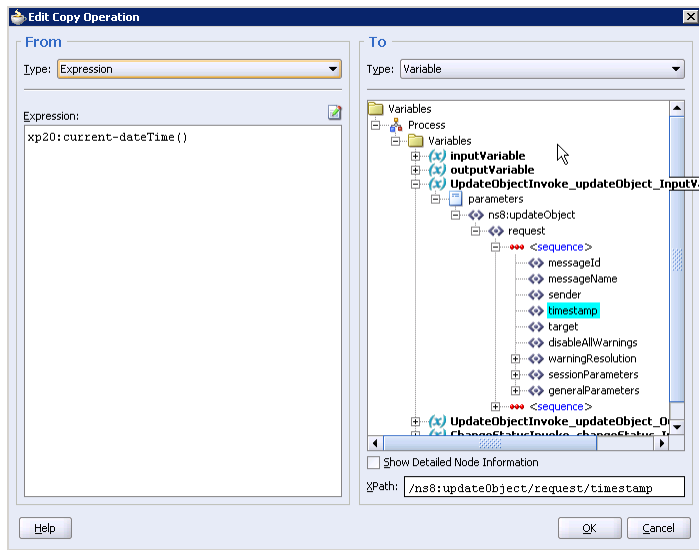


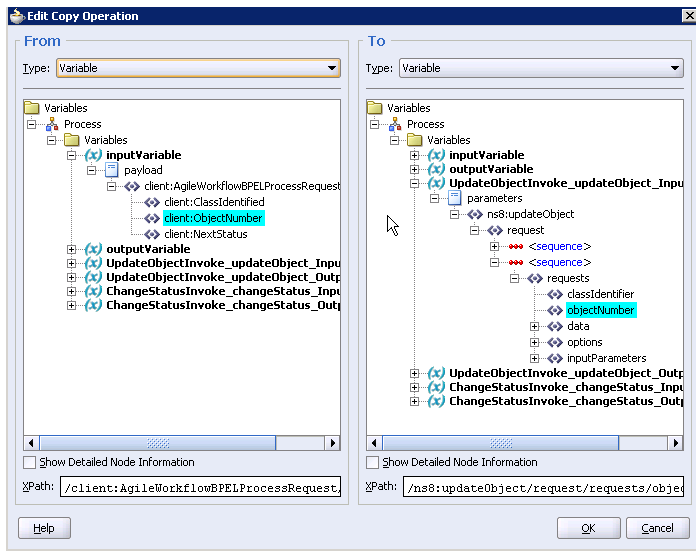
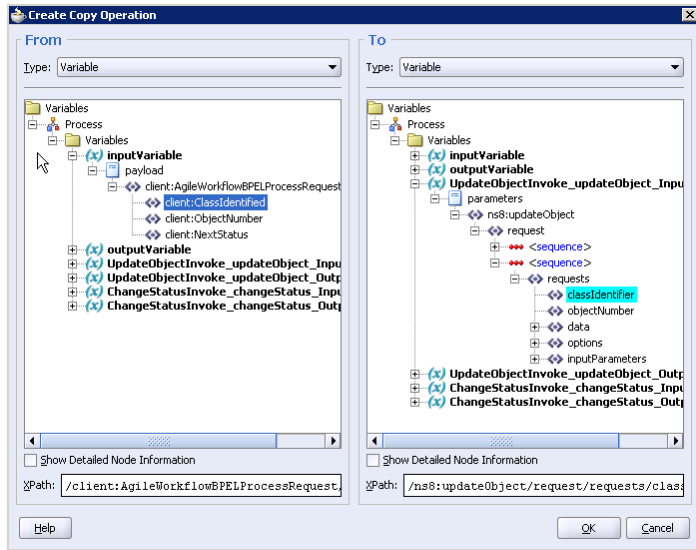


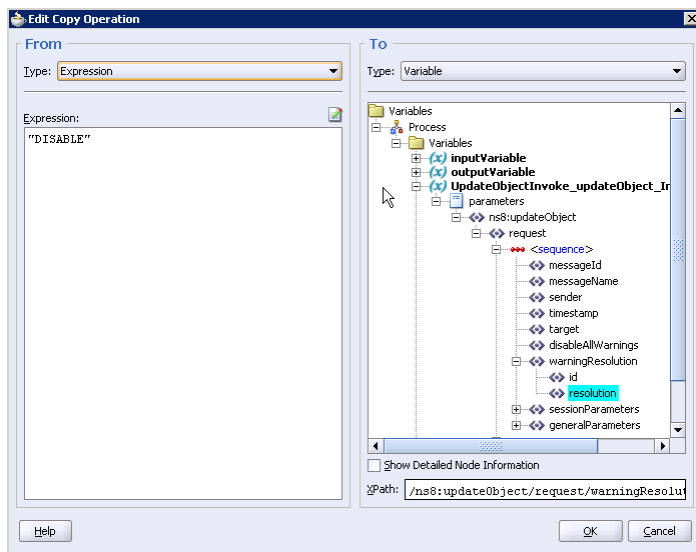
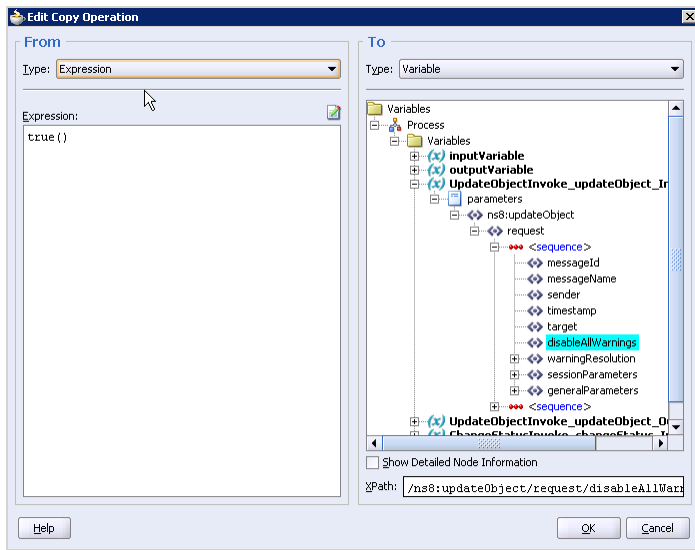


33. Create another Assign object and call it UpdateObjectAssign.

34. Create a few copies of Operations as in ChangeStatusAssign.

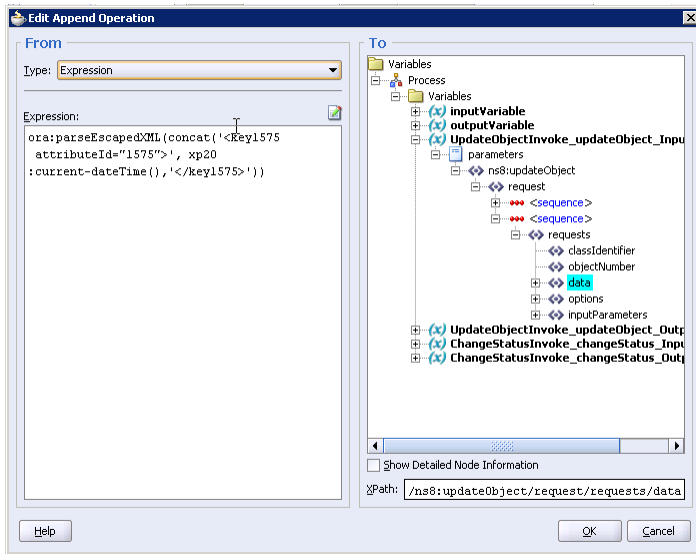




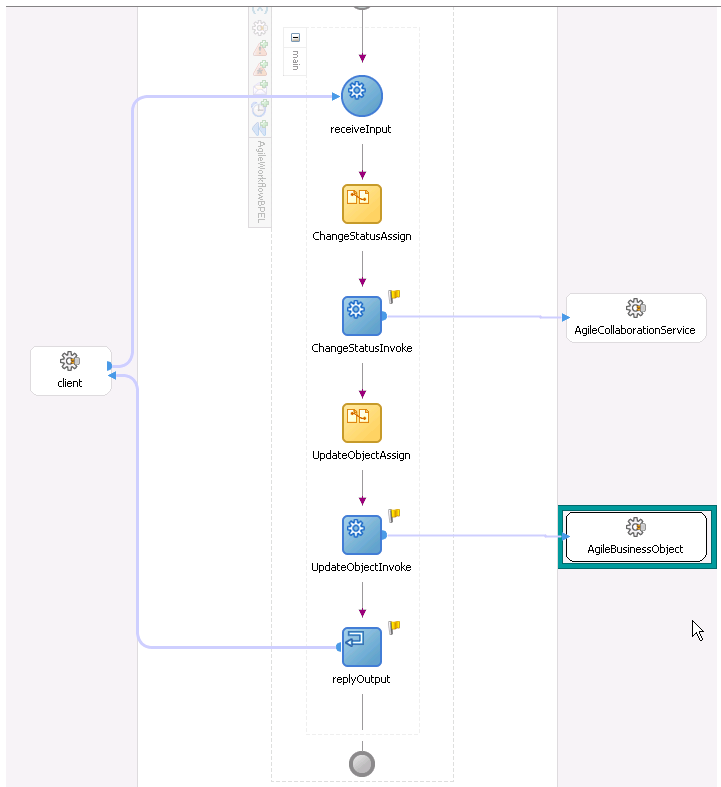


35. Create an append operation.

Note 1575 is the attribute ID used to configure the Agile system.



The completed BPEL process appears.



Deploying and Invoking the BPEL Process in Agile PLM

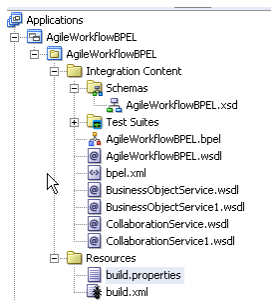
This chapter includes the following:

- Deploying the BPEL Process..... 30
- Configuring Agile PLM and Invoking the BPEL Process 31

Deploying the BPEL Process

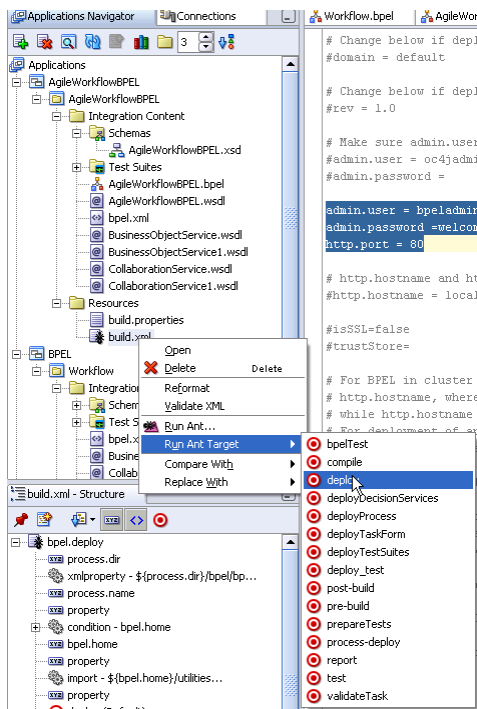
To deploy the BPEL process:

1. Double click the build.properties in the left pane to open this file.



2. In build.properties file, define the following three properties make sure they match your environment.
 - admin.user = bpeladmin
 - admin.password =welcome1
 - http.port = 80

3. Right click build.xml and run deploy target.



This will compile and deploy the BPEL process to the targeted BPEL server.

```

Apache Ant - Log
[deployProcess] Successfully deployed the process "AgileWorkflowBPEL" on server "localhost" and port "80"

deployTaskForm:
[echo]
-----
| Deploying workflow form for AgileWorkflowBPEL on [j2ee.hostname], port 80
-----

[deployTaskForm] There are no forms to deploy

deployDecisionServices:
[echo]
-----
| Deploying decision services for AgileWorkflowBPEL on [j2ee.hostname], port 80
-----

[deployDecisionServices] There are no decision services to deploy

process-deploy:

post-build:

deploy:

BUILD SUCCESSFUL
Total time: 29 seconds
  
```

Apache Ant (Finished)

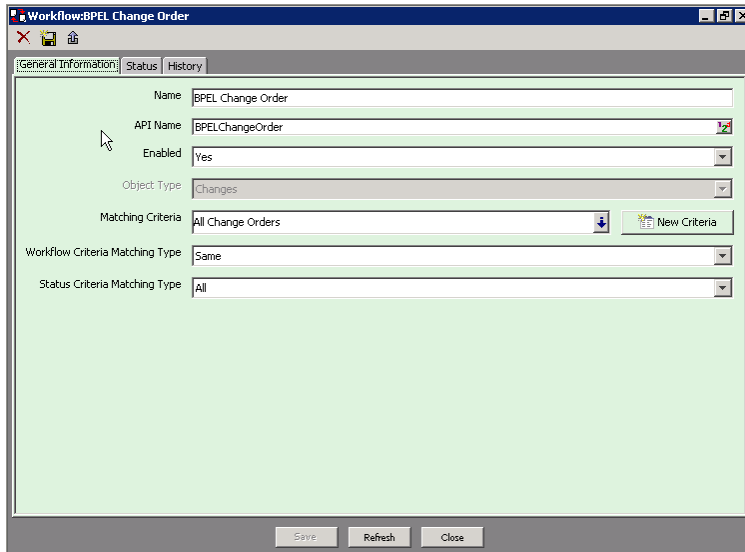
Configuring Agile PLM and Invoking the BPEL Process

This section assumes a working knowledge of the Agile PLM system. In particular, using the Java

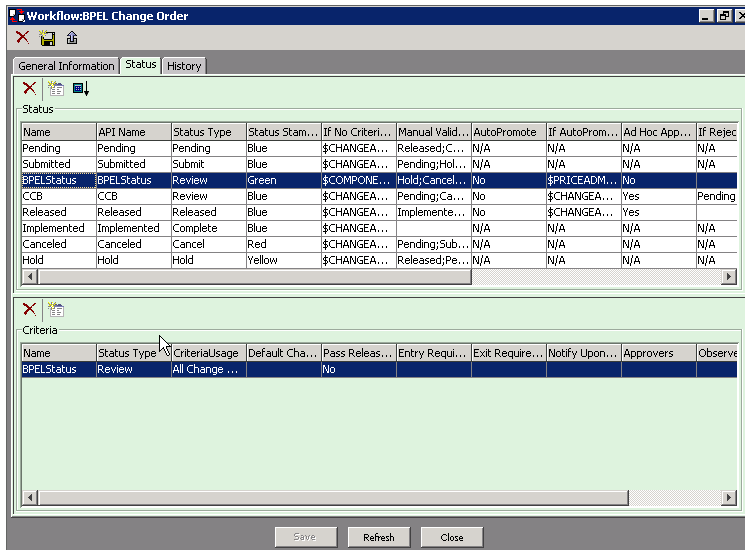
Client to configure Workflow, Event, Subscriber, and Event Handlers!. Refer to the applicable Agile PLM 9.3 Administrator and SDK Developer guides.

To configure Agile PLM:

1. Create a new Workflow called BPEL Change Order. For procedures, refer to *Agile PLM Administrator Guide*.



2. Add a new status for the newly created workflow.



3. Create a Agile 9.3 Event.

The screenshot shows the 'Event:BPEvent' configuration window. The 'General Information' tab is selected. The fields are as follows:

- Event Type: Change Status for Workflow
- Name: BPEvent
- API Name: BPEvent
- Description: (empty)
- Enabled: Yes
- Workflow: BPEvent Change Order
- Object Type: Change Orders
- Status - From: Submitted
- Status - To: BPEventStatus

Buttons at the bottom: Save, Refresh, Close.

4. Create the Event Handler.

The screenshot shows the 'Event Handler:BPEventHandler' configuration window. The 'General Information' tab is selected. The fields are as follows:

- Handler Type: Script PX
- Name: BPEventHandler
- API Name: BPEventHandler
- Description: (empty)
- Role: (empty)
- Enabled: Yes
- Script:


```
import com.agile.agileDSL.ScriptObj.IBaseScriptObj
import com.oracle.xmlinc.Workflow.BPManager
// add other import statements here
void invokeScript(IBaseScriptObj obj) {
//script body starts here.

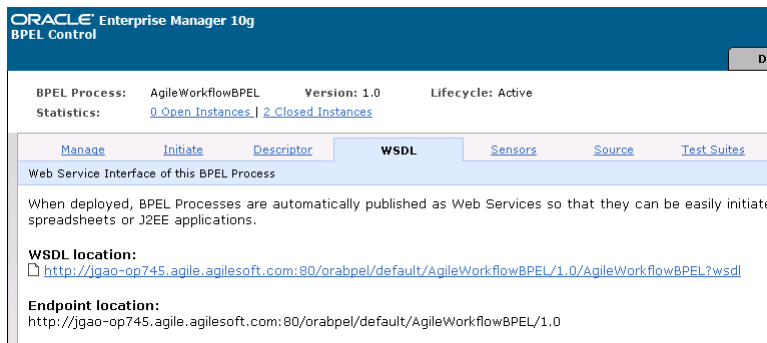
BPManager bpManager = new BPManager()
System.out.println(obj.getObjectClassId())
System.out.println(obj.getObjectNumber())
//System.out.println(ChangeConstants.CLASS_ECO)
bpManager.invoke("ECO", obj.getObjectNumber(), "CCB")
}
```

Buttons at the bottom: Validate Script, Save, Refresh, Close.

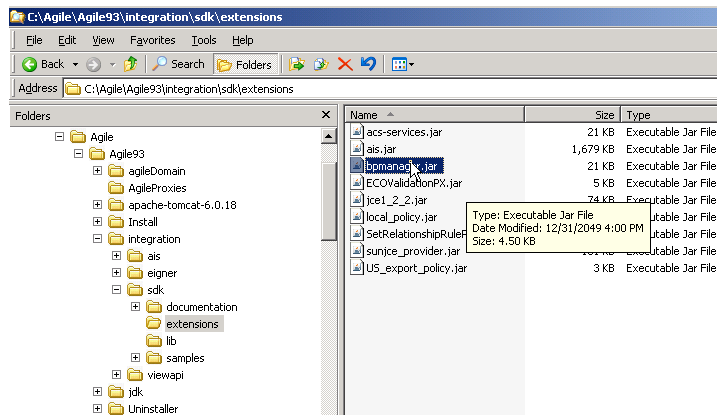
To invoke the BPEL Process:

1. Generate stubs for the above BPEL process using WSDL2Java as shown below.

- a. Log on Oracle BPEL Control and download the WSDL file for the BPEL process



- b. Follow the WSDL2Java tutorial in the apache website to generate stubs <http://ws.apache.org/muse/docs/2.2.0/tutorial/03-wsdl2java.html>
- c. Compile and use Java jar command to zip all the stubs and put it into the extension directory of Agile system.



- d. Restart Agile PLM system
2. In the event handler, write Groovy scripting code to use stubs to invoke the BPEL process.

3. Create event subscriber to associate the event with event handler

The screenshot shows the configuration window for an Event Subscriber. The fields are as follows:

Name	BPELSubscriber
API Name	BPELSubscriber
Description	
Enabled	Yes
Event	BPELEvent
Event Type	Change Status for Workflow
Object Type	Change Orders
Event Handler	BPELHandler
Handler Type	Script PX
Trigger Type	Post
Execution Mode	Synchronous
Order	0
Error Handling Rule	Continue

Buttons: Save, Refresh, Close

Footer: Agile Administrator (admin) | ormi://93Demo.qa.agilesoft.com/Agile | 9.3 (Build 42)

4. Create a change order and set the workflow as BPEL Change Order.

The screenshot shows the configuration window for a Change Order (ECO-00155). The fields are as follows:

Number	ECO-00155
Status	Unassigned
Change Type	ECO
Change Category	
Description of Change	
Reason For Change	
Reason Code	
Workflow	BPEL Change Order
Change Analyst	
Originator	Agile Administrator (admin)
Date Originated	06/15/2009 10:10:42 PM BST
Date Released	

Buttons: Save, Refresh, Close

Footer: Unassigned

5. Route the change from Pending to BPELStatus, it will trigger the event, and then event handler will invoke the business process, the business process in turn calls Agile web services API to route the change from BPELStatus to CCB and logs the timestamp when the business process gets invoked in Agile object field (BPEL Triggered)

