

Oracle OpenWorld 2008

Hands on Lab

**Service Enable Oracle Siebel with Oracle SOA Suite BPEL
Process Manager**

Table of Contents

1	TUTORIAL OVERVIEW	3
2	CREATING THE BPEL PROCESS.....	3
	2.1 Startup	3
	2.2 Creating the BPEL Workspace and Project	4
	2.3 Creating a Partner Link to Insert the Siebel Account	8
	2.4 Creating Invoke Activity	11
	2.5 Creating an Initial Assign to Send Input Request Data to the Invoke	15
	2.6 Creating an Assign to Send Output Data from the Invoke	17
	2.7 Validating, Compiling and Deploying the BPEL Process	19
	2.8 Example of BPEL Process	20
3	TEST THE INTEGRATION.....	21
	3.1 Testing the BPEL Process from the BPEL Console.....	21
4	ADDITIONAL CREDIT	25
	4.1 BPEL Console	25
	4.2 JDeveloper.....	25
	4.3 Log into Siebel and Generate the WSDL	25

1 Tutorial Overview

This lab demonstrates native web services integration to Siebel using Oracle BPEL Process Manager. Siebel (since 7.5.3) has increased support for native web services, and exposes a number of functions as Application Service Interfaces (ASIs). There is also tooling support in Thin Client (browser) and Siebel Tools to generate and consume WSDLs. In this tutorial you will Query a Siebel Account.

For more hands-on tutorials, check out the Oracle By Example Tutorials on the Siebel and Fusion Middleware Best Practice Center on Oracle Technology Network [<http://www.oracle.com/technology/tech/fmw4apps/siebel>].

In this Lab:

- You will generate a WSDL in Siebel for the Siebel Account Business Service*
- Consume the WSDL in BPEL as a Partnerlink
- Invoke the QueryByID() operation within the WSDL
- Pass in ID and the expected response will be Account details.

We have completed step (a) for you, and the generated WSDL is located in your FMW4Apps → LabFiles → Siebel folder. This was done primarily to ensure we finish the lab in the short duration. For those of you who complete the lab and still have time follow instructions in end of this doc on steps to generate the WSDL.

Software Used:

- Siebel 8.0 Call Center
- Oracle SOA Suite version 10.1.3.1 [available for download from [OTN](#)]
- Oracle JDeveloper version 10.1.3.4 [available for download from [OTN](#)]

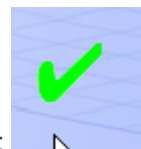
2 Creating the BPEL Process

2.1 Startup

Start **Oracle SOA Suite** using the StartBPEL shortcut in FMW4Apps → Shortcuts Folder on your desktop.



When it completes you will see a check mark like this:

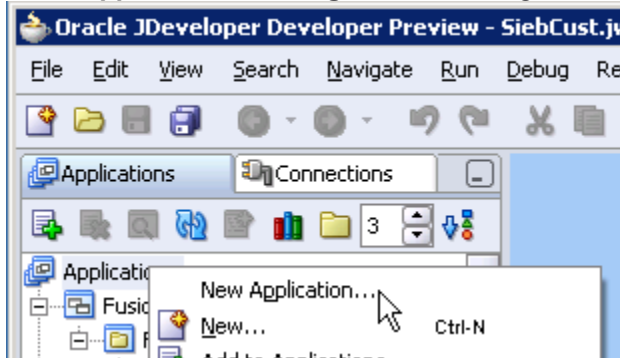


Then start **Oracle Jdeveloper** by clicking on the icon as below in the same folder.

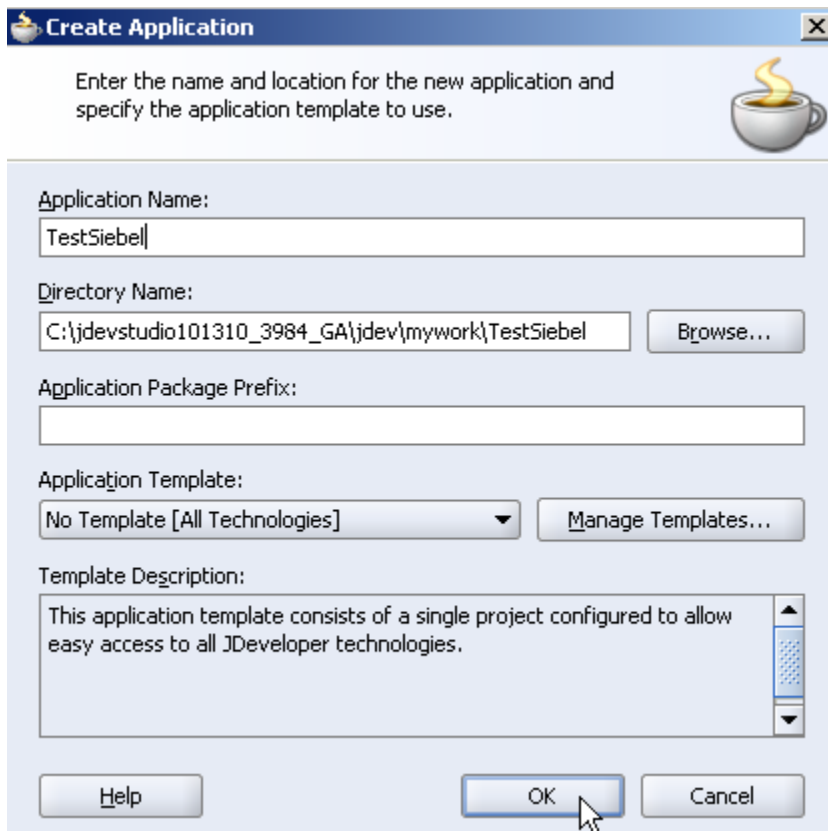


2.2 Creating the BPEL Workspace and Project

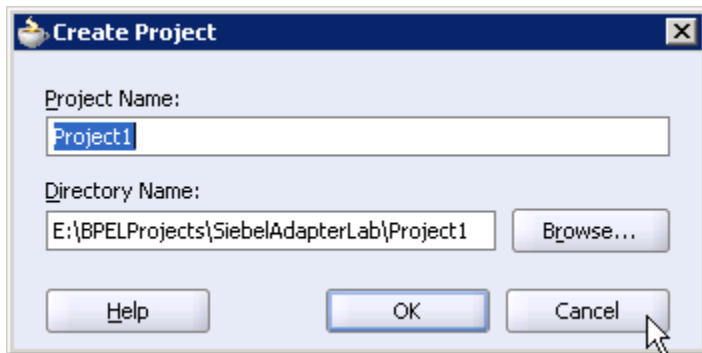
In the **Applications – Navigator** window, right-click on **Applications** and select **New Application...**



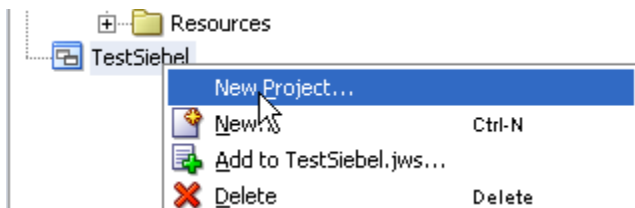
Name your application. In this example, we'll use **TestSiebel**. Your directory names may be different. Take the defaults as-is. Click OK to create the application.



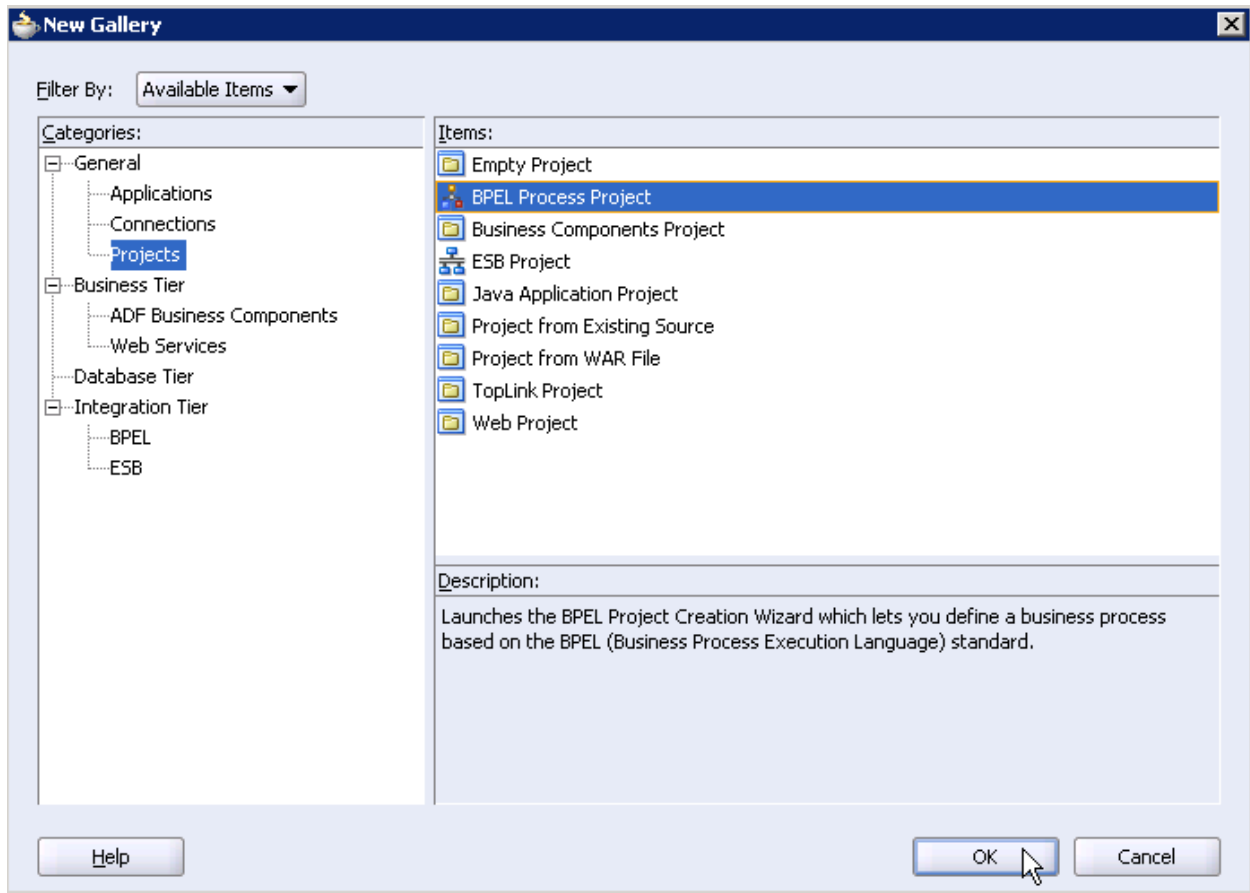
A Create Project window will appear. Click Cancel. (Yes Click Cancel).



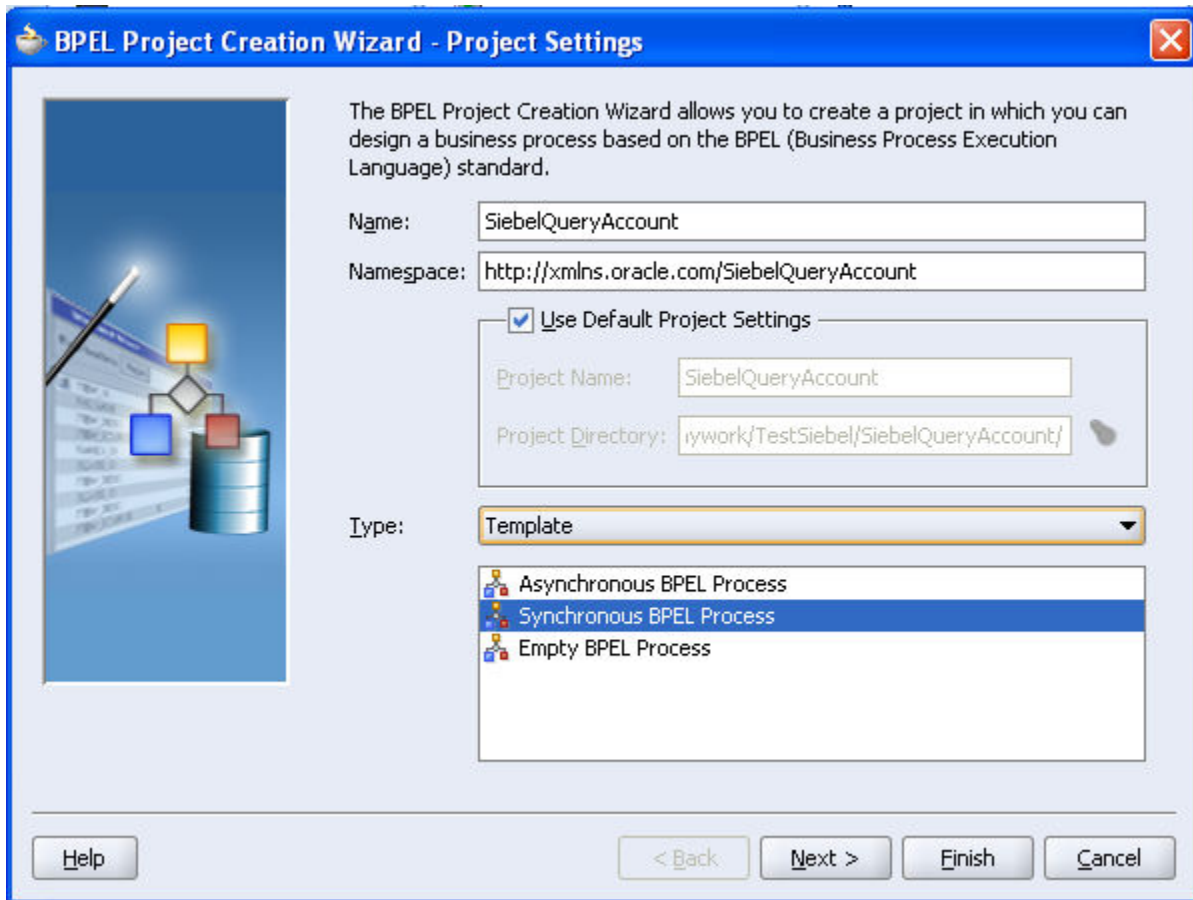
You will now be back at the main JDeveloper window, and you will see your **TestSiebel** in the applications window. Right click on the TestSiebel application, and select New Project...



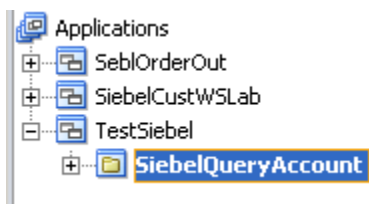
In the New Gallery window, select BPEL Process Project and click OK.



Name the process **SiebelQueryAccount** in the Name field and select **Synchronous BPEL Process** as the template. Click Finish.

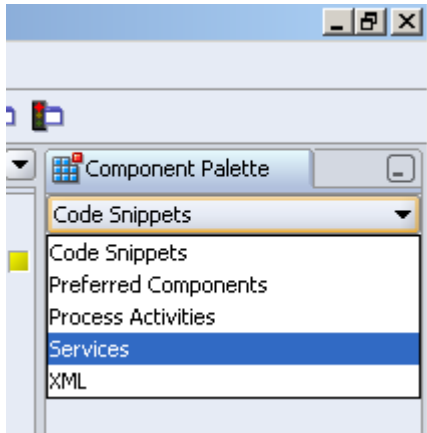


The SiebelQueryAccount project is then created under your TestSiebel application. You can see the files in the **Applications – Navigator** window.

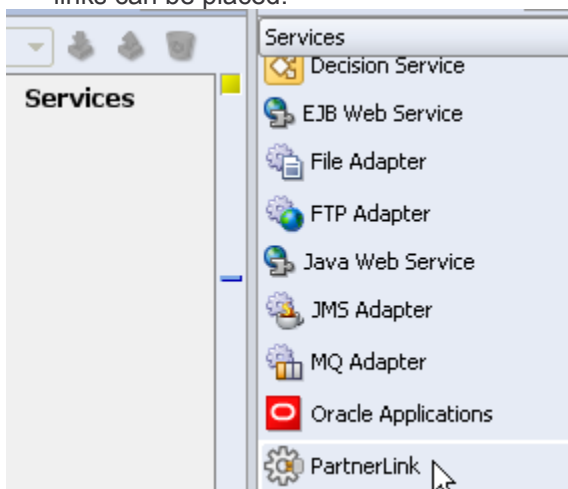


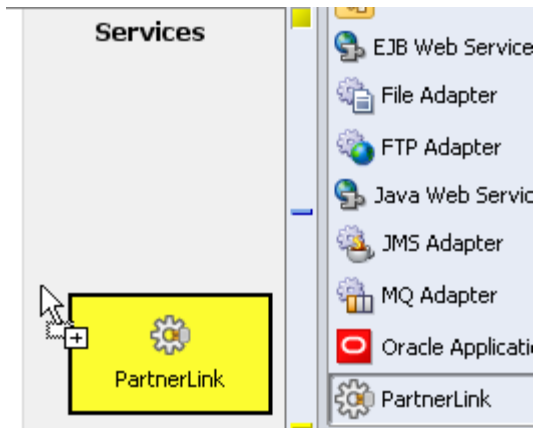
2.3 Creating a Partner Link to Insert the Siebel Account

1. Select **Services** from the drop-down list of the **Component Palette** section in the upper right section of the JDeveloper Designer.

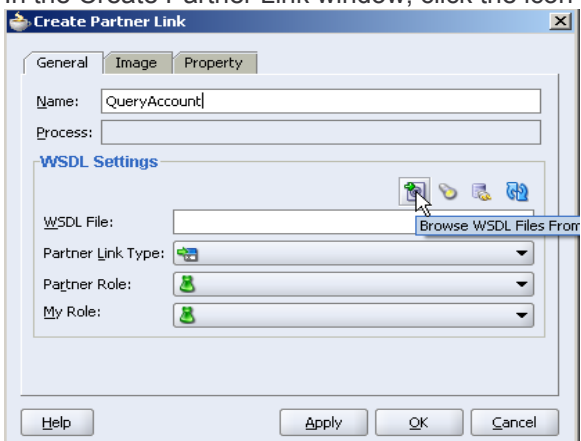


2. Drag and drop a **PartnerLink** activity onto the **right** side of the designer window anywhere beneath the Services header. Note that both sides of the designer window have sections in which partner links can be placed.

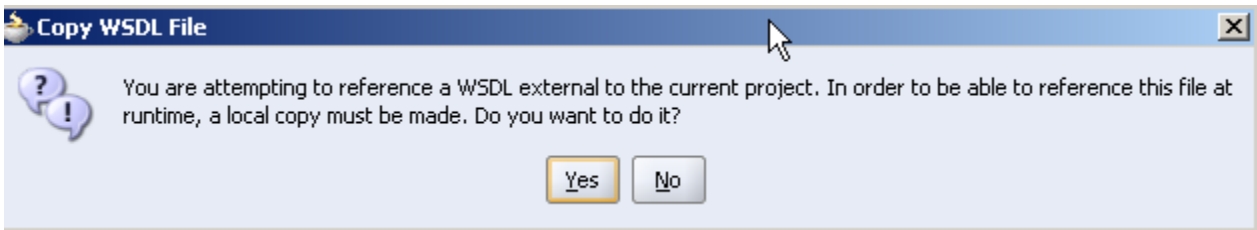




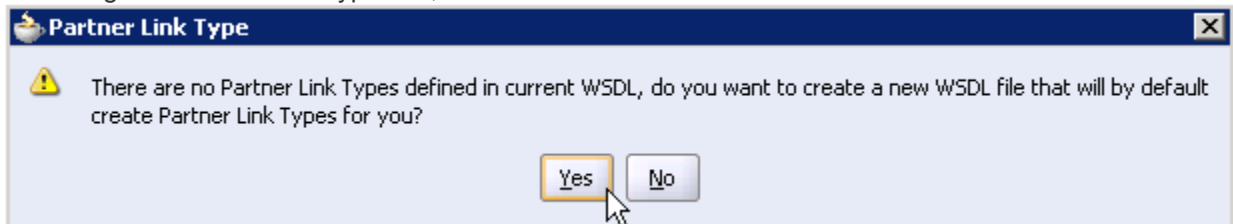
3. In the Create Partner Link window, click the icon **Browse WSDL Files From Local File System**.



4. Browse and select the Siebel WSDL file SiebelAccount80.wsdl. It will be located in your desktop folder FMW4Apps → LabFiles



You will get a Partner Link Type box, select Yes.



Enter the following values to finish creating the partner link
Name: **QueryAccount**
Partner Role: **CustAccount_Role**
Partner Link Type is automatically selected.
Click **Ok**

Create Partner Link

General Image Property

Name: QueryAccount

Process:

WSDL Settings

WSDL File: |/SiebelQueryAccount/bpel/SiebelAccount80Ref.wsdl

Partner Link Type: CustAccount_PL

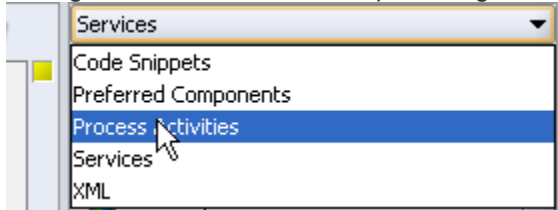
Partner Role: CustAccount_Role

My Role: ----- Not Specified -----

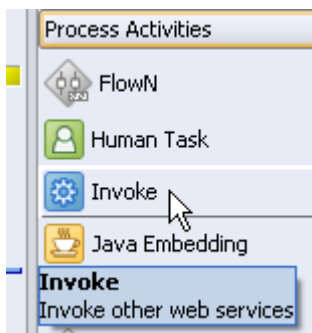
Help Apply OK Cancel

2.4 Creating Invoke Activity

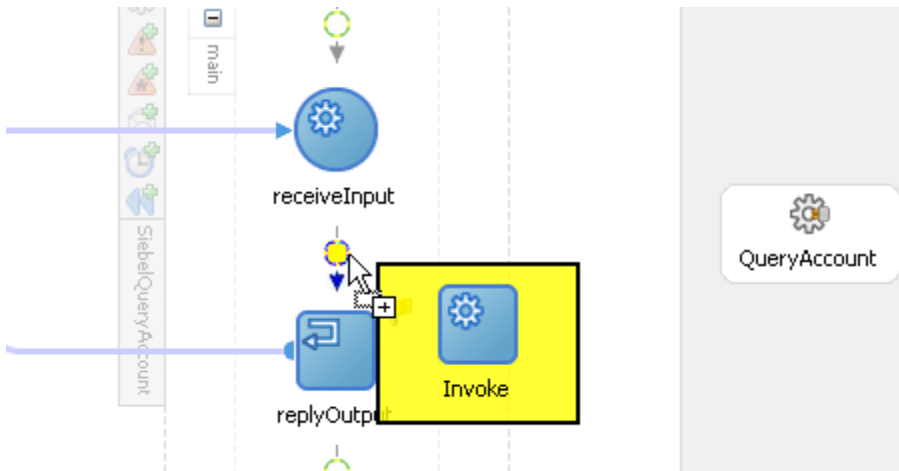
1. Select **Process Activities** from the drop-down list of the **Component Palette** section in the upper right section of the JDeveloper Designer.

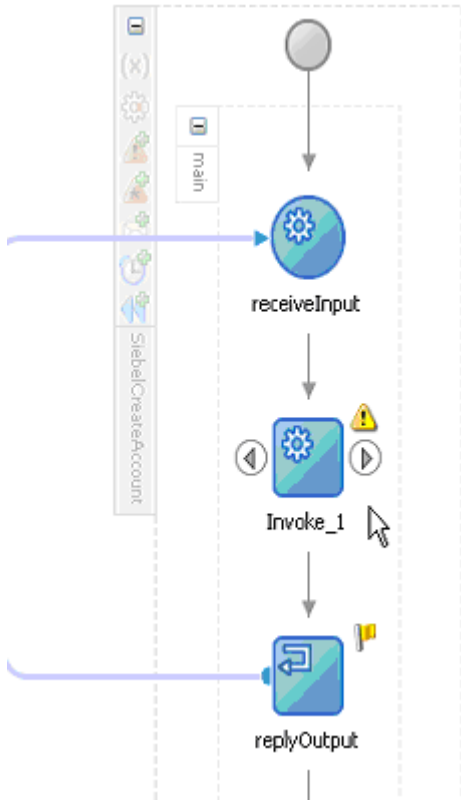


2. Drag and drop an **Invoke** from the Component Palette section.



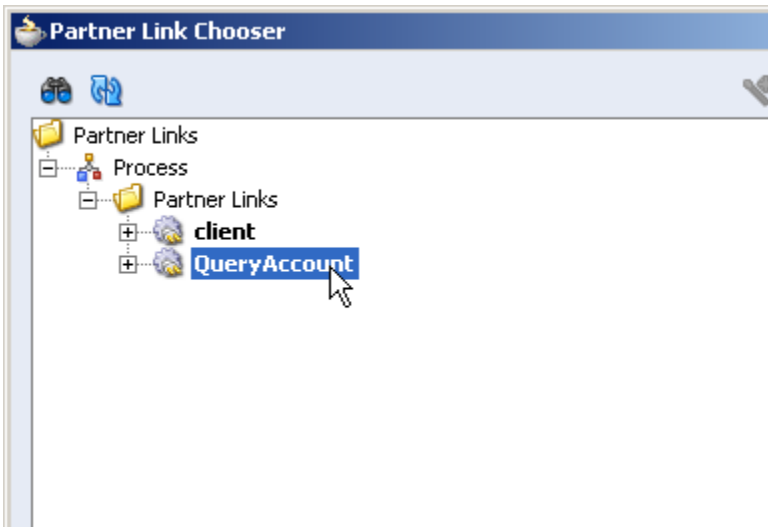
And drop onto the canvas immediately **after** the **receiveInput** activity



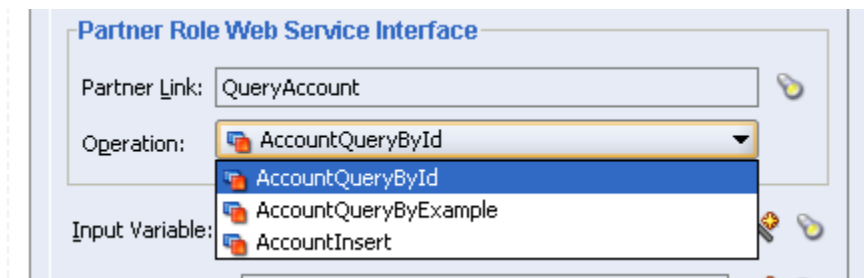


3. Double-click the **Invoke_1** icon to display the Invoke window

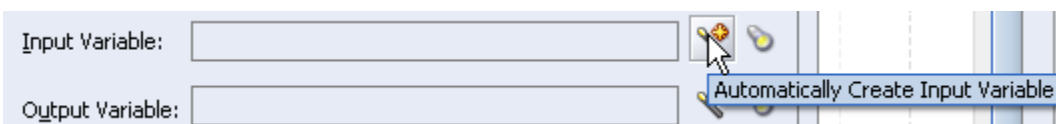
4. Enter the following values.
 - a. Set Invoke Name to **Invoke_QuerySiebelAccount**
 - b. Set Partner Link to **QueryAccount** by selecting the flashlight icon clicking OK.



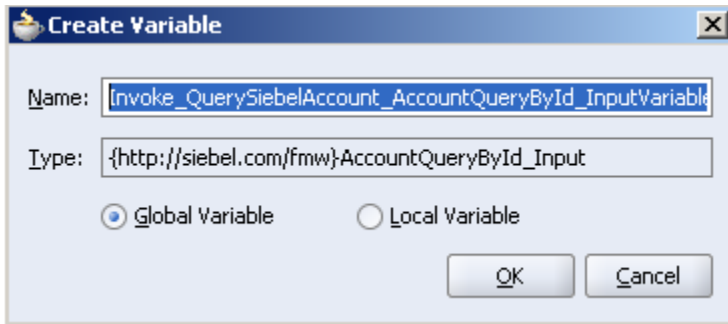
- c. Set the Operation to **AccountQueryById**



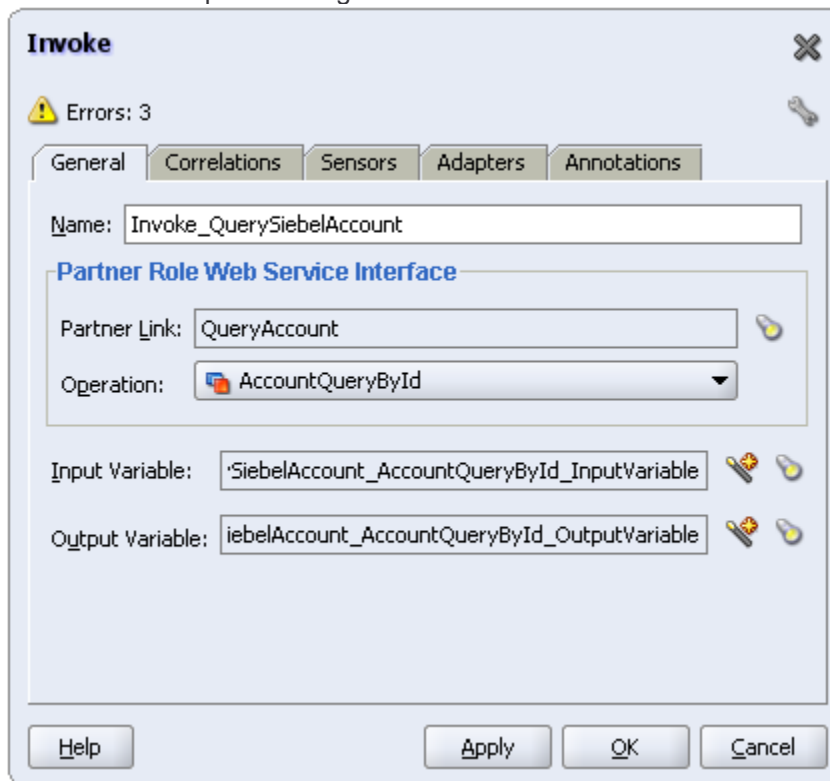
- d. Click on the '**Auto-create variable**' magic wand icon next to the **Input Variable** text field to create a variable for this activity.



Click **OK** to accept the default variable name.

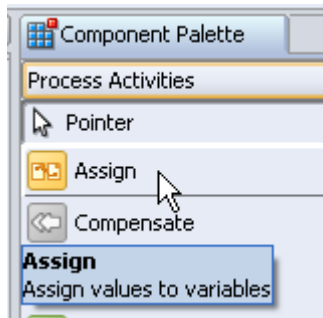


- e. Click on the 'Auto-create variable' magic wand icon next to the **Output Variable** text field to create a variable for this activity. Click **OK** to accept the default variable name, as done for previous step.
- f. Click **OK** to accept the changes to Invoke. Your invoke should look like this.

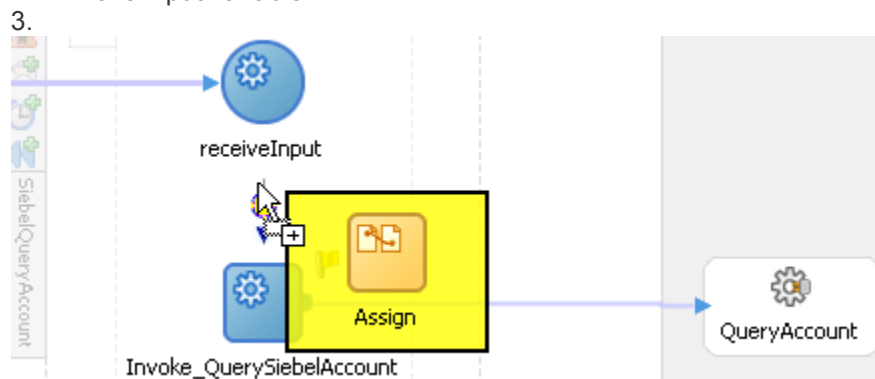


2.5 Creating an Initial Assign to Send Input Request Data to the Invoke

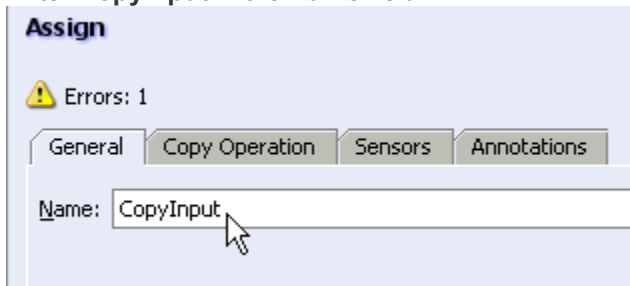
1. Verify **Process Activities** is still selected from the drop-down list of the **Component Palette** section in the upper right section of the JDeveloper Designer.



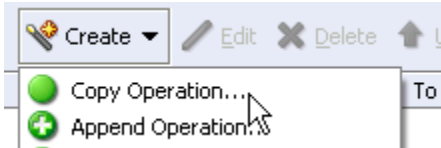
2. Drag and drop an **Assign** activity from the Component Palette section between the **receiveInput** and **Invoke_QuerySiebelAccount** activities so we can copy the **input** field in the input Variable to the Invoke input variable.



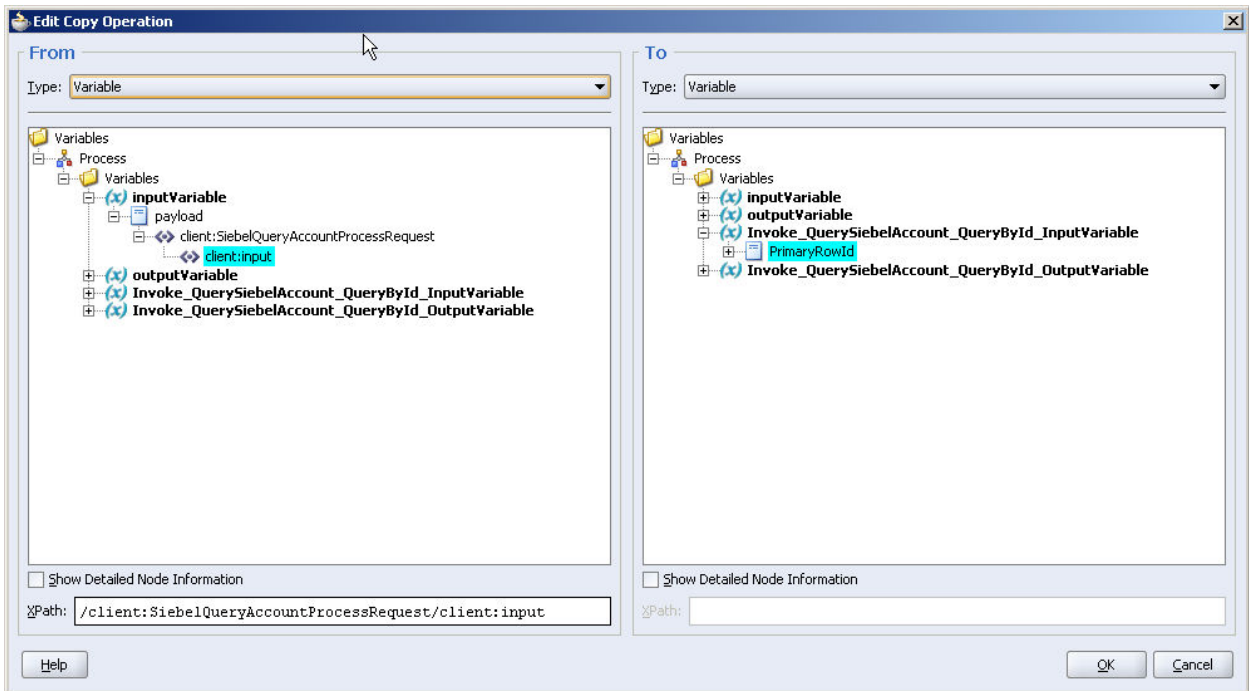
4. Double-click the **Assign** icon that you just dropped to display the Assign window.
5. Click the **General** tab
6. Enter **CopyInput** in the **Name** field



7. Click **Apply**
8. Click the **Copy Operation** tab
9. Click **Create** to display the Create options window, and then select Copy Operation...

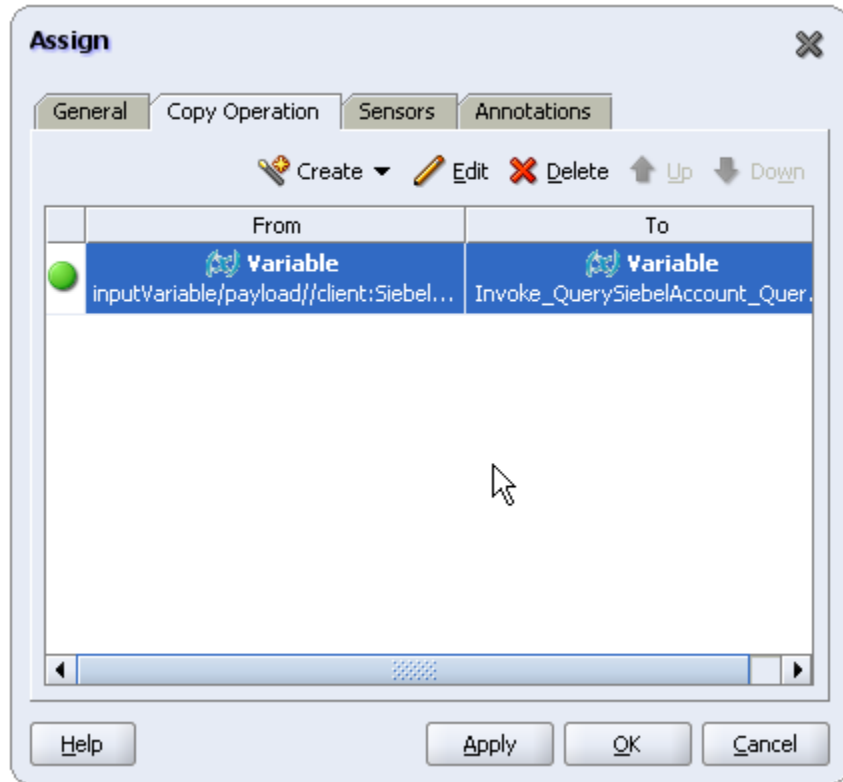


10. Make the changes to appear as follows:



Note that we are taking the user input to the BPEL process and assigning it to the input variable of the Invoke step in BPEL.

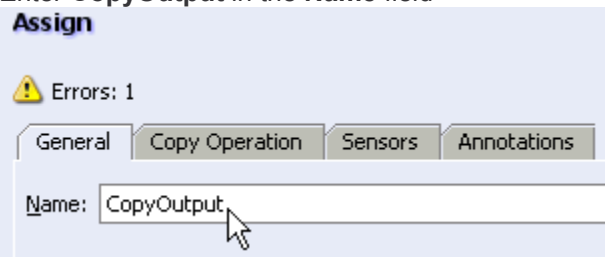
11. Click **OK** to close the Create Copy Rule window and the Assign window



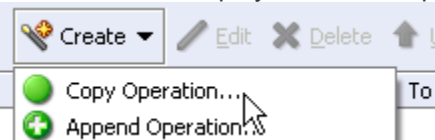
2.6 Creating an Assign to Send Output Data from the Invoke

Now, we will repeat the steps above to set the response from Siebel Query call to the output variable. This time the Assign will be **AFTER** the **Invoke**.

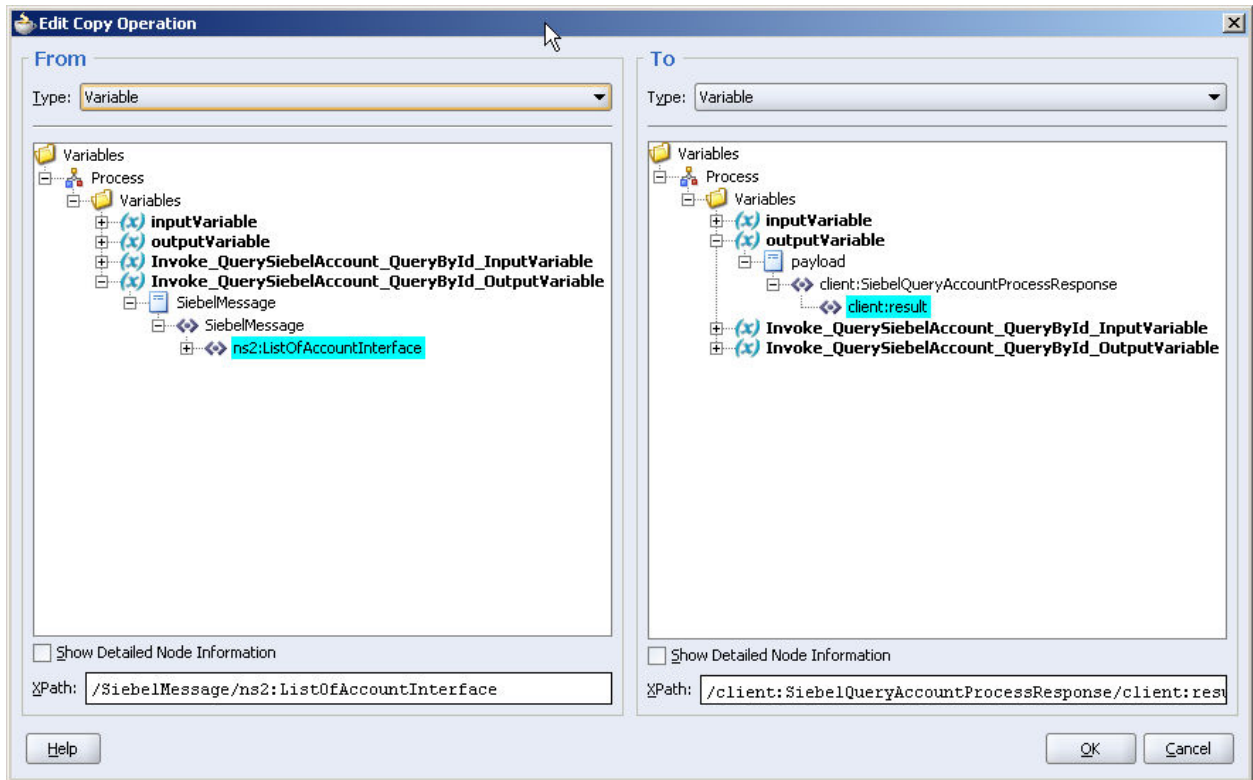
1. Verify **Process Activities** is still selected from the drop-down list of the **Component Palette** section in the upper right section of the JDeveloper Designer.
2. Drag and drop an **Assign** activity from the Component Palette section between the **Invoke_QuerySiebelAccount** and **replyOutput** activities so we can copy the **output** field in the Invoke output Variable to the replyOutput input variable.
3. Double-click the **Assign** icon that you just dropped to display the Assign window.
4. Click the **General** tab
5. Enter **CopyOutput** in the **Name** field



6. Click **Apply**
7. Click the **Copy Operation** tab
8. Click **Create** to display the Create options window, and then select Copy Operation...



9. Make the changes to appear as follows:



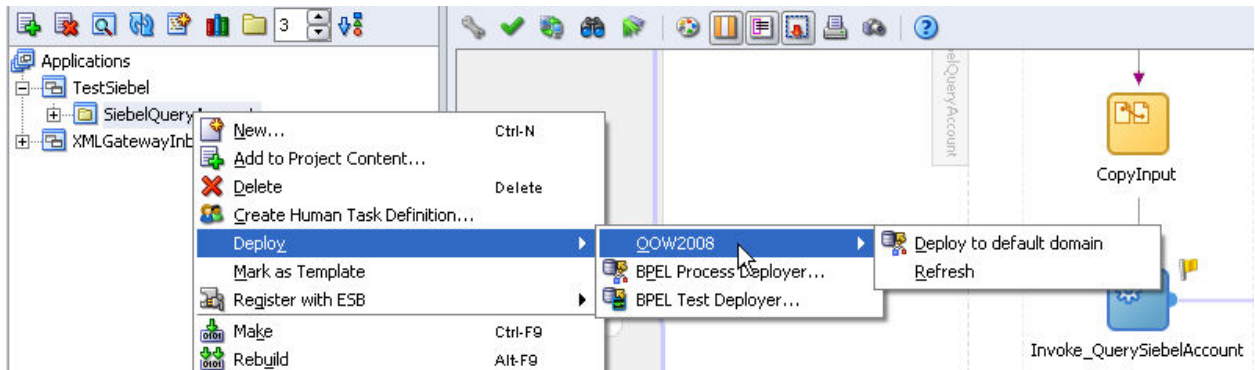
Note here that we are taking the output of the Invoke step and assigning it to output variable of the bpel process. This is result that user will see returned in BPEL console.

10. Click **OK** to close the Create Copy Rule window and the Assign window. At this point you are done building your BPEL process. Now lets build and deploy this.

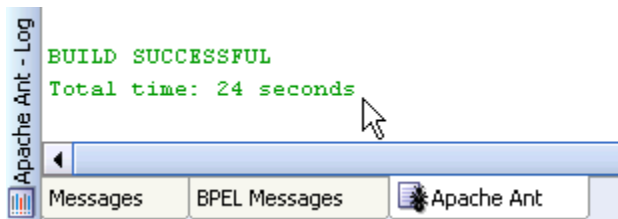
2.7 Validating, Compiling and Deploying the BPEL Process

Note: For this section you need to ensure you have BPEL manager started and a link to the default domain from Jdeveloper.

1. Go to the **Applications Navigator** section
2. Right-click the **SiebelQueryAccount** BPEL Project
3. Select **Deploy** → **OOW2008** → **Deploy to default domain**

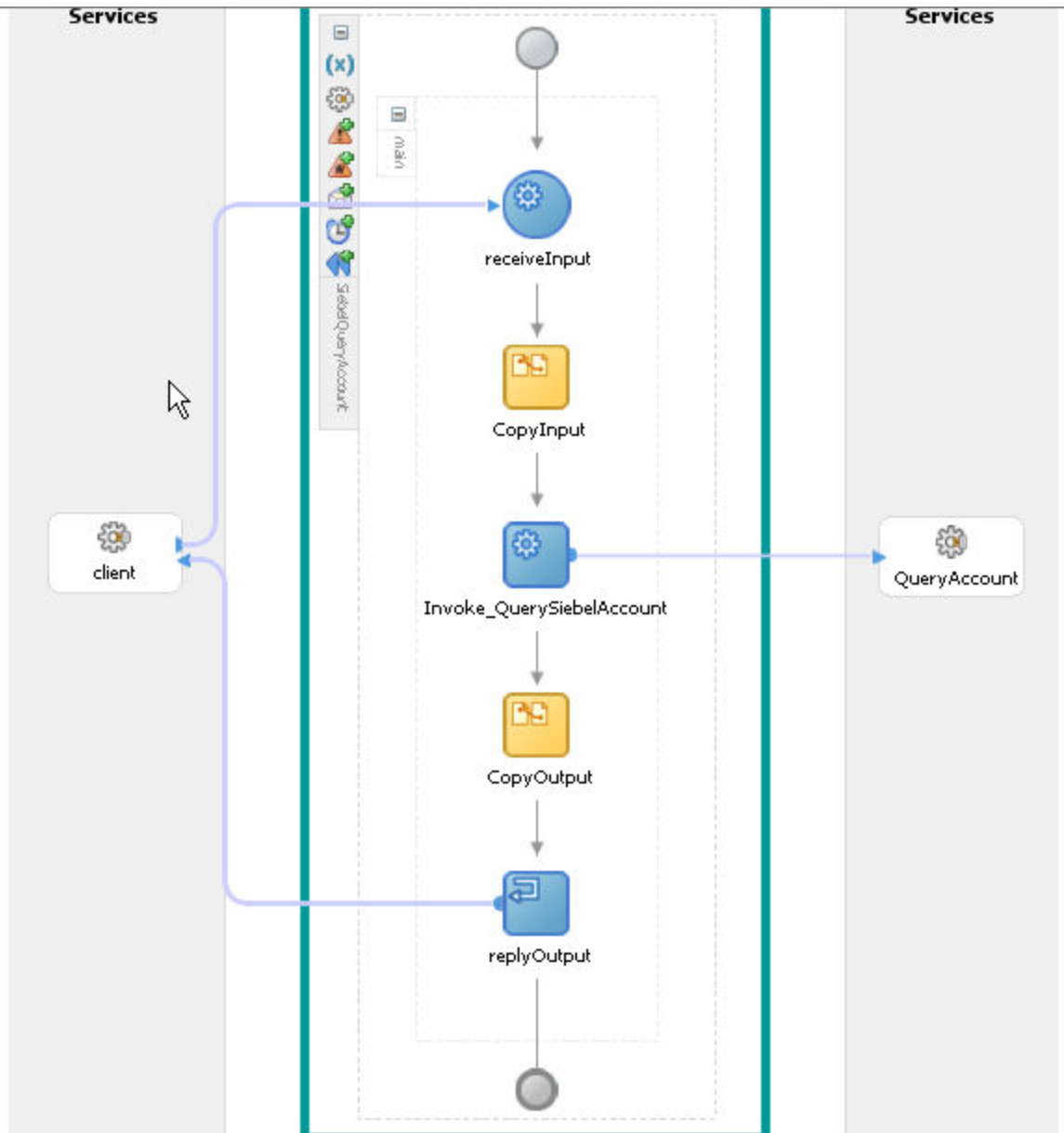


4. This compiles and deploys the BPEL process. Review the bottom window for any errors. If there are no errors, deployment was successful.



5. If you have errors, make appropriate corrections and deploy again. Ignore any warnings.
6. Once you deploy with no errors, your workflow project is now ready for testing.

2.8 Example of BPEL Process



3 Test The Integration

3.1 Testing the BPEL Process from the BPEL Console

You can test that your BPEL project works by initiating an Insert from the BPEL Console.

Log into the Oracle BPEL Console using Internet Explorer by clicking on the URL on your shortcuts folder



1. The default Username is **oc4jadmin** and the default password is **welcome1**

The login screen for the Oracle BPEL Console. At the top, it says 'ORACLE APPLICATION SERVER 10g'. Below that, 'BPEL CONSOLE' is displayed in large white letters. There are two input fields: 'Username:' with 'oc4jadmin' entered, and 'Password:' with 'welcome1' entered (masked with dots). A 'Login' button is at the bottom right.

2. Click on **SiebelQueryAccount** in the Deployed BPEL Processes list.

The Oracle Enterprise Manager 10g BPEL Control Dashboard. The header is blue with 'ORACLE Enterprise Manager 10g BPEL Control' and a 'Dashboard' button. Below the header is a table with two main sections: 'Deployed BPEL Processes' and 'In-Flight BPEL Process Instances'.

Deployed BPEL Processes		In-Flight BPEL Process Instances	
Name		Instance	BPEL Process
SiebelQueryAccount			
TaskActionHandler			
TaskManager			

3. This takes you to the Initiate tab of the SiebelQueryAccount BPEL Process where you will see a HTML form. Here enter a Siebel Record ID to Query. Sample Siebel Account Record Id is

1-5GZO

Other samples to use are
1SIA-7UKI
24-28V6

Enter one of these in the payload input as shown below. Then click on Post XML Message button.

[Manage](#) **Initiate** [Descriptor](#) [WSDL](#) [Sensors](#)

Testing this BPEL Process

Initiating a test instance

To create a new 'test' instance of this BPEL Process, fill this form and click on 1

Operation HTML Form XML Source

WS-Security Include In Header

WS-Addressing Include In Header

payload

input xsd:string

Note: XML source view contents will not be reflected in the HTML form view

Save Test

Perform stress test

- You should see a message saying the test request was processed synchronously, and have the time that it took to finish. It will also give you the return result which is the customer Account details.

ORACLE Enterprise Manager 10g Manage BPEL
BPEL Control

Dashboard BPEL Processes Instances

BPEL Process: SiebelQueryAccount **Version:** 1.0 **Lifecycle:** Active
Statistics: [0 Open Instances](#) | [2 Closed Instances](#)

[Manage](#) **Initiate** [Descriptor](#) [WSDL](#) [Sensors](#) [Source](#) [Test Suites](#)

Test Instance Initiated

Your test request was processed synchronously. It took 15.703seconds to finish and generated the following Value:

```
<SiebelQueryAccountProcessResponsehttp://xmlns.oracle.com/SiebelQueryAccount>
  <resulthttp://www.siebel.com/xml/Account%20Interface>
    <Accounthttp://www.siebel.com/xml/Account%20Interface>
      <AccountId>1-5GZ0</AccountId>
      <AccountStatus>Active</AccountStatus>
      <CompetitorFlag>Y</CompetitorFlag>
      <CreditAutoApprovallLimit>0</CreditAutoApprovallLimit>
      <CurrencyCode>USD</CurrencyCode>
      <NumberOfEmployees>300</NumberOfEmployees>
      <Location>San Francisco</Location>
      <MainFaxNumber>4157424550</MainFaxNumber>
      <MainPhoneNumber>4157424500</MainPhoneNumber>
      <Name>AG Edwards & Sons, Inc</Name>
      <ParentAccountId>1-5GZE</ParentAccountId>
      <ParentAccountLocation>HQ-Distribution</ParentAccountLocation>
      <ParentAccountName>AG Edwards & Sons, Inc</ParentAccountName>
      <PartnerFlag>N</PartnerFlag>
      <PrimaryOrganization>Millennium Healthcare HC ENU</PrimaryOrganization>
      <SkipCreditCheck>N</SkipCreditCheck>
      <Type>Commercial</Type>
      <ListOfBusinessAddress>
```

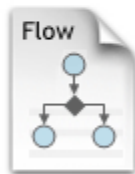
- You can now click on either the Audit Instance link to verify the BPEL process completed, and look at some of the values at each step. Here is an example looking at the Audit link.

```

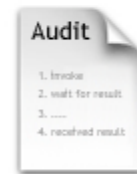
</ListOfRelatedIndustry>
</Account>
</result>
</SiebelQueryAccountProcessResponse>

```

For more information:



[Visual Flow](#)



[Audit Instance](#)

A page shown below comes up. You can click on various operations in the page to see each step in the BPEL Process that queried Siebel.

ORACLE Enterprise Manager 10g BPEL Control Manage BPEL Domain | Logged to

Dashboard **BPEL Processes** **Instances**

Title:	Instance #50003 of SiebelQueryAccount	Last Modified:	3/19/08 5:38:21
Reference Id:	50003 Tree Finder	State:	closed.complete
BPEL Process:	SiebelQueryAccount (v. 1.0)	Priority:	0

[Manage](#) [Flow](#) **Audit** [Debug](#) [Interactions](#) [Sensor Values](#) [Test](#)

Audit trail of this BPEL instance | [View Raw XML](#) [As of 3/19/08 5:39:17 P

[2008/03/19 17:38:05] New instance of BPEL process "SiebelQueryAccount" initiated (# "50003").

```

<process>
  <sequence>
    <receiveInput>
      [2008/03/19 17:38:05] Received "inputVariable" call from partner "client" More...
    <copyInput>
      [2008/03/19 17:38:05] Updated variable "Invoke_QuerySiebelAccount_AccountQueryById_InputVariable" More...
    <invokeQuerySiebelAccount>
      [2008/03/19 17:38:21] Invoked 2-way operation "AccountQueryById" on partner "QueryAccount". More...
    <copyOutput>
      [2008/03/19 17:38:21] Updated variable "outputVariable" More...
    <replyOutput>
      [2008/03/19 17:38:21] Reply to partner "client". More...
  </sequence>
</process>

```

Click on "more" link as shown above to see additional details on process execution, such as input and output parameters.

4 Additional Credit

4.1 BPEL Console

- a) Go back to the BPEL Dashboard and look at the completed process instances.
- b) Check the executed flow diagram of completed instances. Click on each step in the process to see the XML data flowing through the step.

4.2 JDeveloper

- c) Explore the different activities you can use in a BPEL process. Look at Process Activities and you will find an array of nodes to use. Interesting ones are Human Task flow, which can be used to include human approval steps and assign tasks to people. Another classic node is the Transform node which is used to create XSLT transformations. It brings up a graphical mapper from which you can drag and drop.

4.3 Log into Siebel and Generate the WSDL

- d) Login to Siebel using the shortcut on the desktop folder (FMW4Apps -> Shotcuts) with userid and password SADMIN/SADMIN
- e) Click on menu item Navigate → Site Map.
- f) Then click on Administration-Web Services
- g) Then click on Inbound Web Services
- h) In the list of Inbound Web Services find the one we used – see below. Use the mouse to find by scrolling down (do not use query button). This is to ensure you don't change the definition of the Web service. (*Note: Siebel saves on change without prompting user.*) You can hit the generate WSDL button and you will be prompted to save. Here choose open to view the WSDL in Internet Explorer. Examine the WSDL for schema types and the server port it is binding to. Do not save. Close the browser.

Inbound Web Services			
Namespace	Name	Status	Comment
http://siebel.com/asi/	Siebel Case Information	Active	
http://siebel.com/asi/	Siebel Contact	Inactive	
> http://siebel.com/fmw	Siebel Customer Account	Active	For Fusion Middleware / Oracle SOA Suite
http://siebel.com/asi/	Siebel Employee	Inactive	

Scroll down and see the service ports and operations we have for this web service. You will find three operations (AccountQuerybyID, AccountQueryByExample and AccountInsert).

Service Ports | Menu ▾ | New Delete Query 1

Name	Type	Business Service, Transport	Address	Binding
> CustAccount	Business Service	Siebel Account HTTP	http://139.185.119.101/eai_enu/start.swe?SWExtSource=WebService&SOAP_RPC_LITERAL	

Operations | Menu ▾ | Add Delete Query 1

Name	Method Display N	Authentication T	Request Filter Se	Request Filter Me	Response Filter S	Response Filter	Method Display
> AccountInsert	Insert	None					
AccountQueryById	Query by ID	None					
AccountQueryByExample	Query by Example	None					

These are out of the box operations exposed on **any** Siebel Business Object. Other operations found on business objects include Update, Upsert, Synchronize and Delete. We have chosen not to expose them in our case. Instructions on how to make these available as operations can be found in Siebel Bookshelf or you can see a tutorial for this on FMW Best Practice Center for Siebel on OTN.

Make sure you have made no changes in the Siebel instance. Logout from the application using File→Logout menu in Siebel browser. Close the browser.