

# Exercise 11

*Creating a BPEL flow*

## Prior Knowledge

*Understand WSDL and Services*

## Objectives

Understand the basics of the BPEL specification, and be able to create and execute a business process using the BPEL tooling in Eclipse. Deploy the BPEL into the WSO2 BPS and be able to track instances etc.

## Software Requirements

- Java Development Kit 7
- WSO2 Developer Studio 3.7
- WSO2 BPS 3.2.0
- WSO2 AS 5.2.1 running the Starbucks OMS service from the previous lab

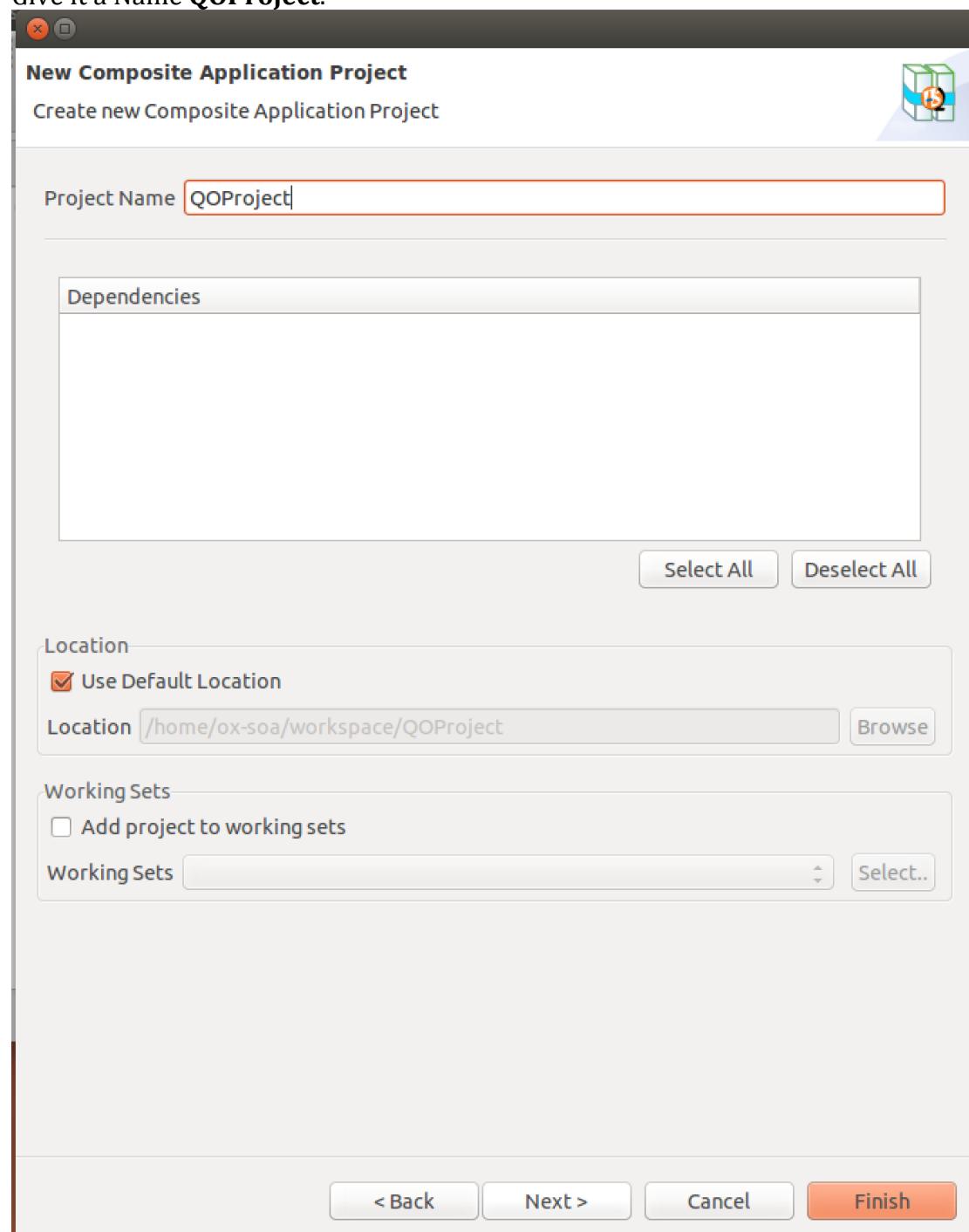
1. In Eclipse, Hit Command-N/Ctrl-N to pull up the New dialog.



© Paul Fremantle 2012. Licensed under the Creative Commons 3.0 BY-SA (Attribution-Sharealike) license.  
See <http://creativecommons.org/licenses/by-sa/3.0/>

## 2. Choose Composite Application Project

Give it a Name **QOProject**.

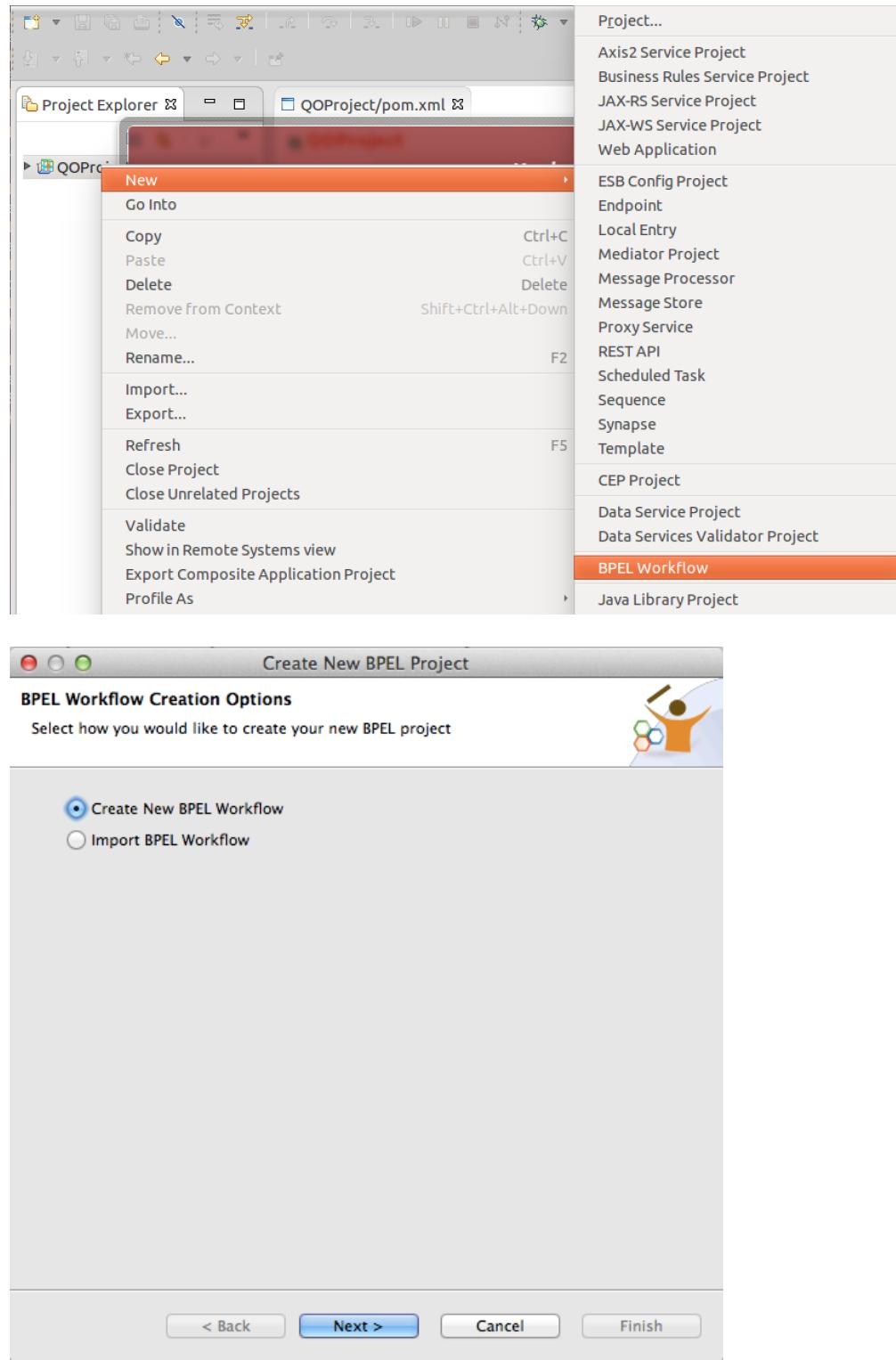


3. Click **Finish**

4. Select **QOProject** and right-click. Select New->BPEL Workflow



5. Select Create New BPEL Workflow:



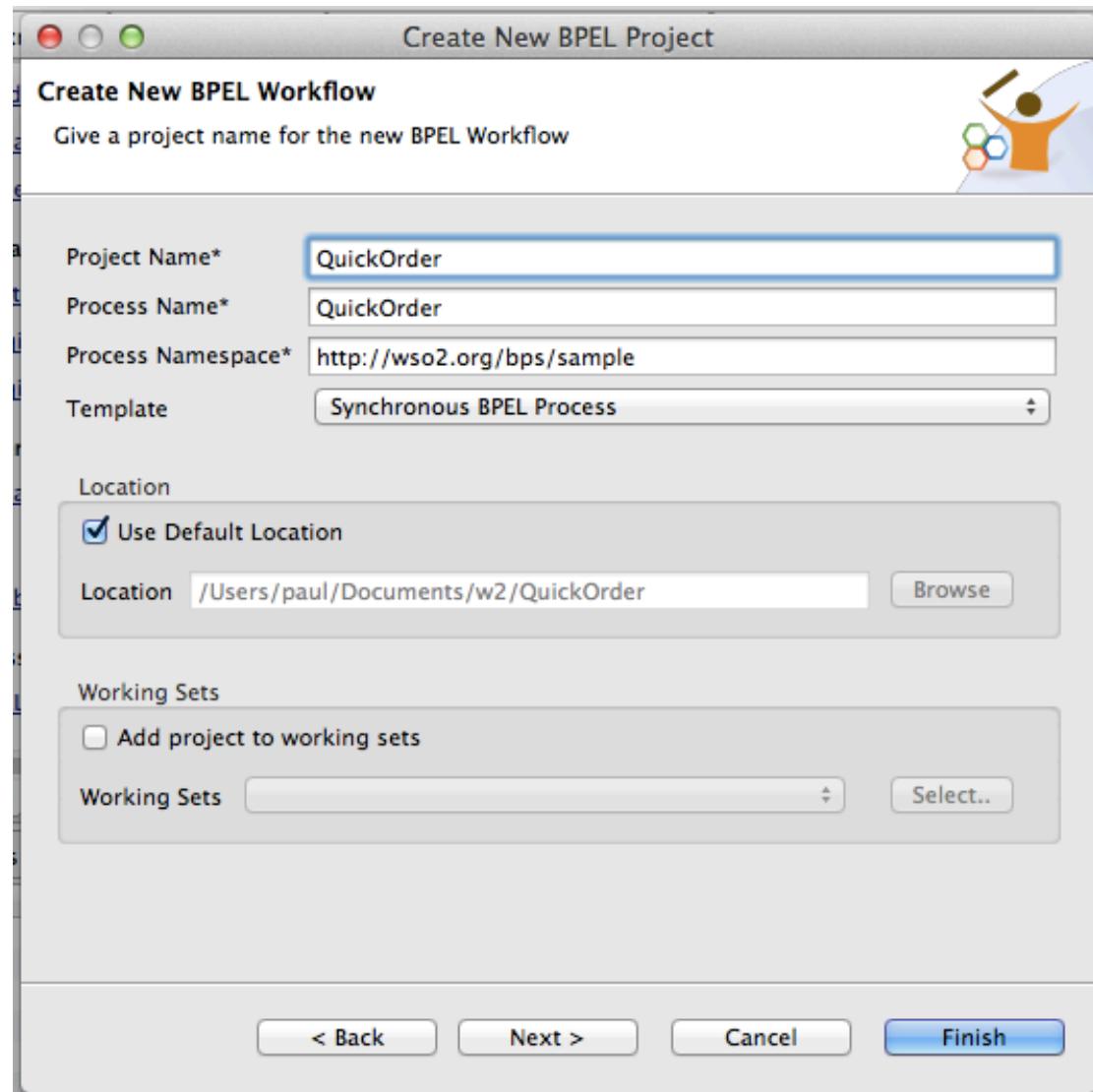
6. Use:

Project Name: **QuickOrder**

Process Name: **QuickOrder**

Template: **Synchronous BPEL Process** [note this is NOT the default!]



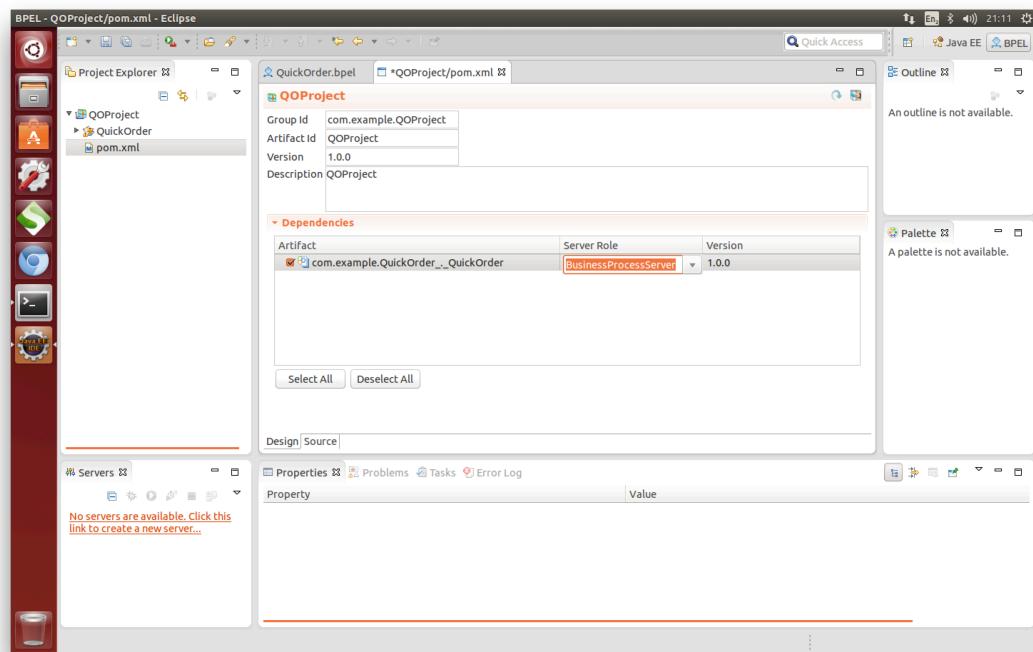


7. Click Finish
8. You may be asked if you want to use the BPEL perspective. You do.



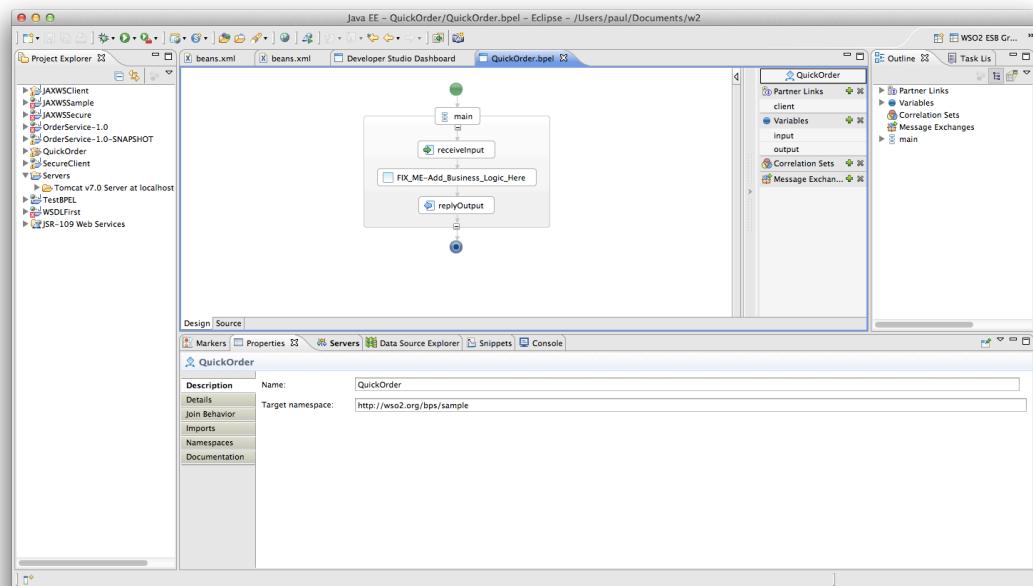
9.

10. Now go back to the “owner” project of QOProject and double-click on the pom.xml file. Select the check box next to the BPEL process:

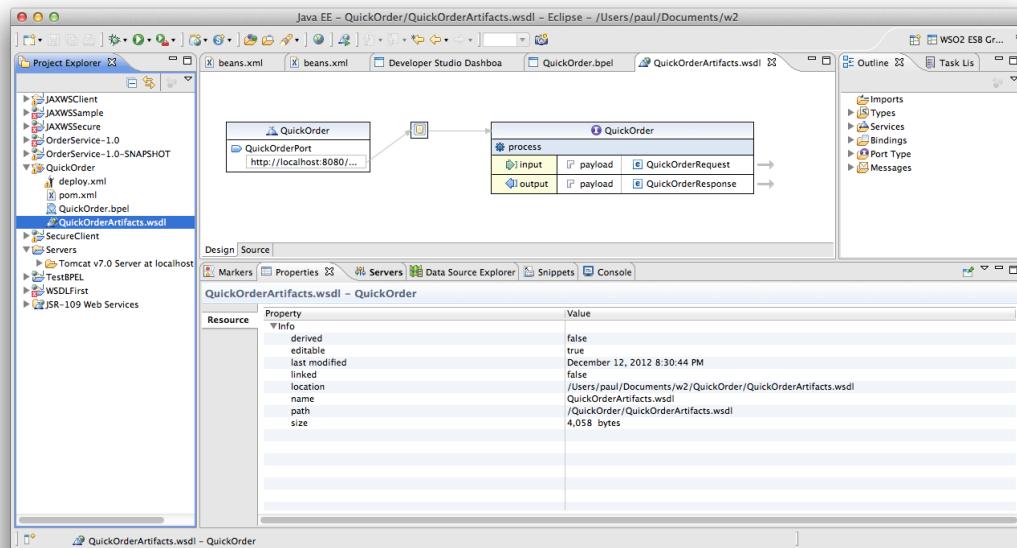


11. Now go back to the QuickOrder.bpel

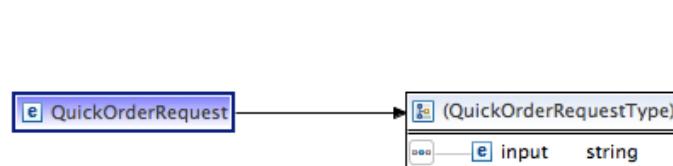
12. Your screen should look similar to this:



13. For the moment ignore the beautiful flow diagram. Instead, edit the QuickOrderArtifacts.wsdl



14. Click on the arrow next to QuickOrderRequest. This will edit the schema for this operation.



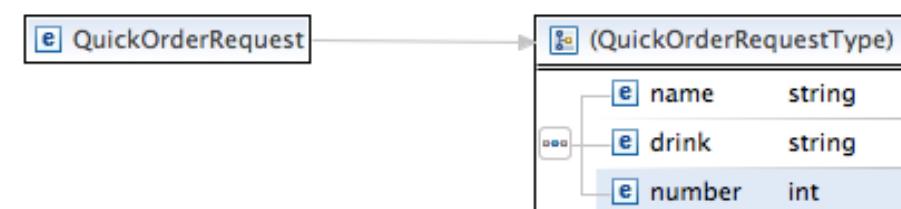
15. Click on the word “input” and rename it to “name”

16. Now Right Click and Insert Element->After

17. Change the name of the NewElement to drink

18. Add another new element after. Make it an int and call it number

19. Now it should look like this:



20. Hit Command-S/Ctrl-S to save.

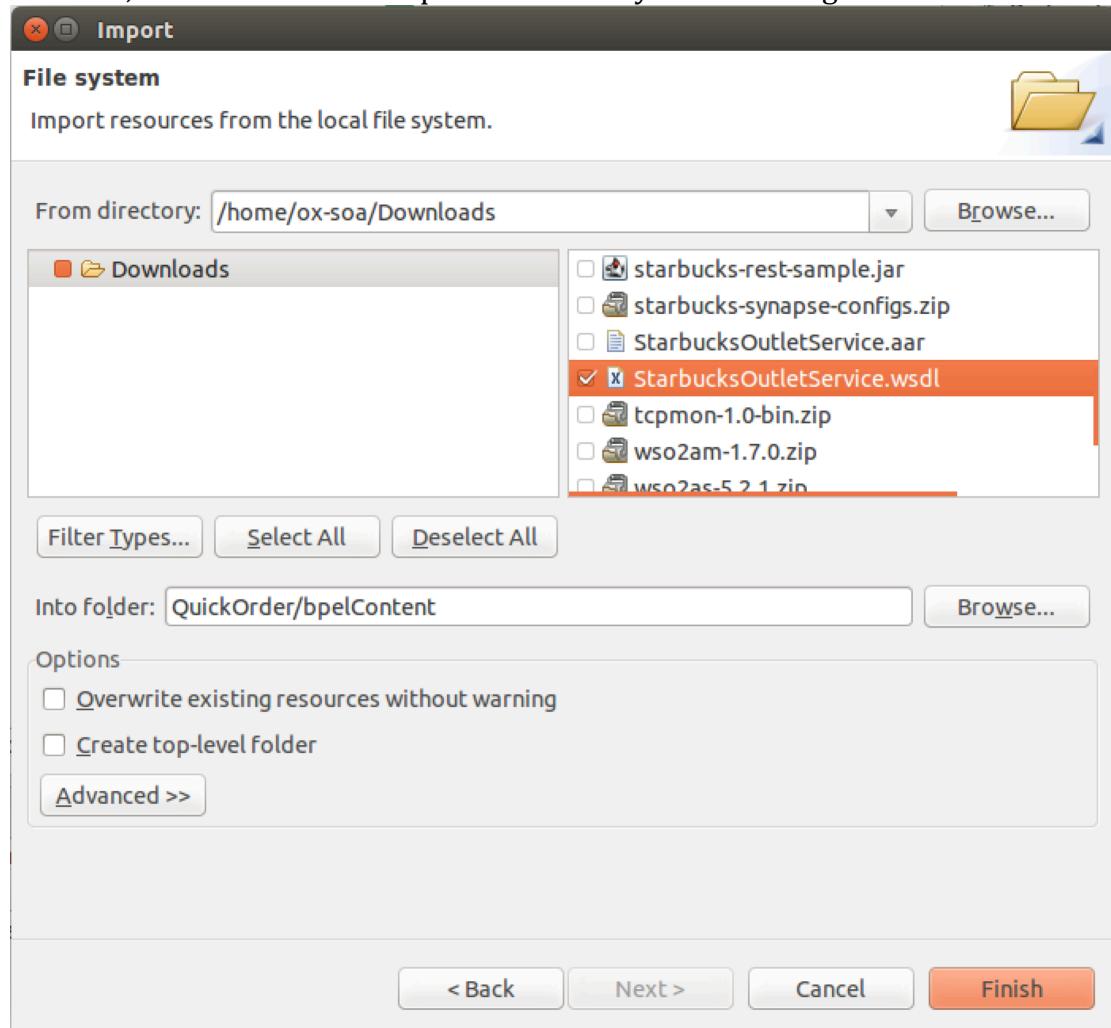


21. Close the Inline Schema tab and the WSDL tab
22. Go back to the Flow Diagram / BPEL page.
23. We are now going to import the Starbucks WSDL.
24. Make sure the AppServer is running and the **Starbucks WSDL** is available using the AppServer console on <https://localhost:9443>
25. Browse the WSDL, and download it to your local file system. Make sure its called .wsdl (not .xml)



26. Now import it into the QuickOrder project **bpelContent** section. Select **bpelContent** using a right-click then select **Import**. Then **File System**

The Eclipse file imported window is a bit weird and pretty hard to describe, but here at least is a picture of what you are aiming for!



27. Click **Finish**

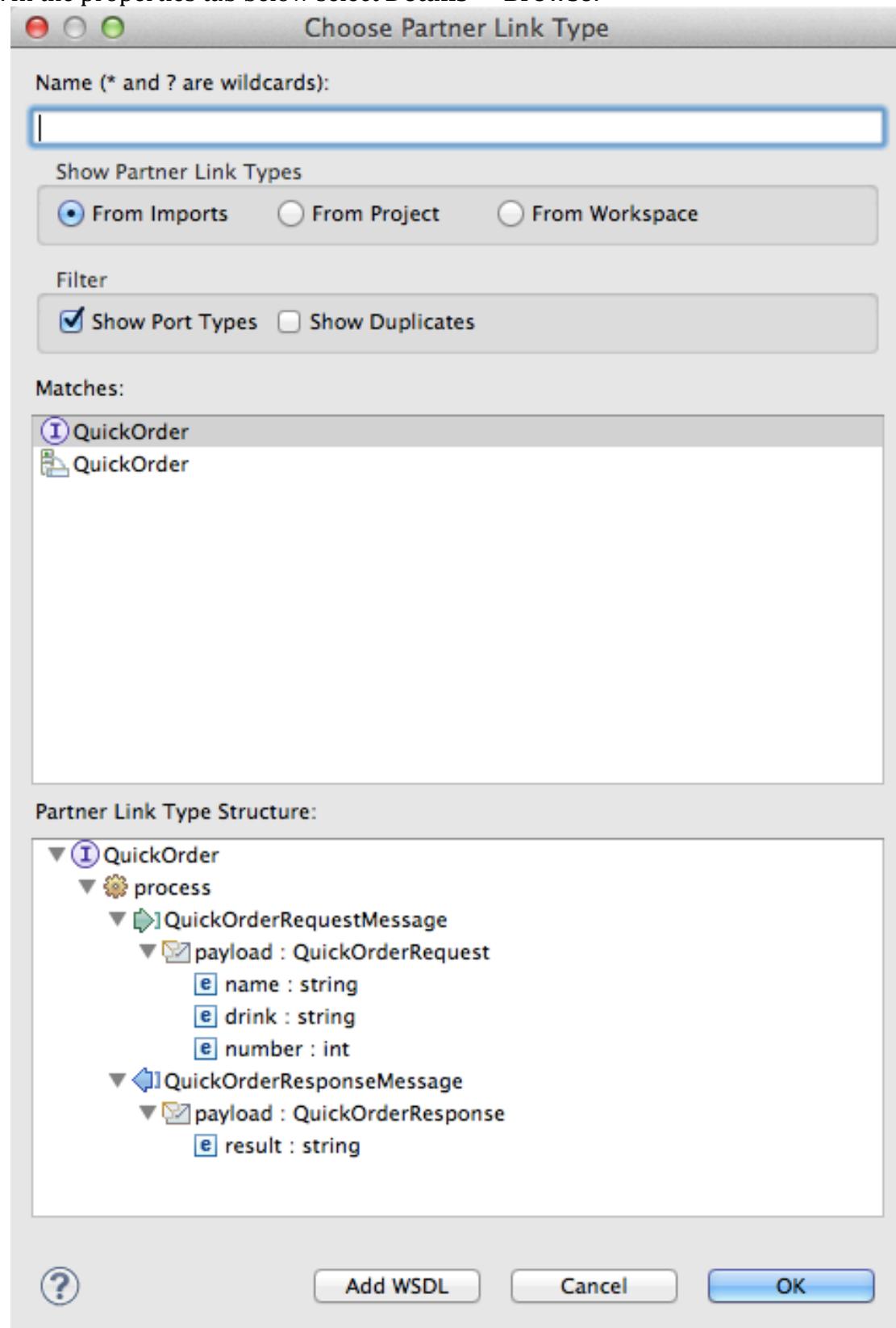
28. Click on the + next to Partner Links also on the right hand side.

29. Change the name of the Partner Link to CoffeeProvider. Note the edit box is a bit weird ☺

30. *BIG HINT. Whenever editing the Properties in the sections below, it is very helpful to double click maximize the properties window.*



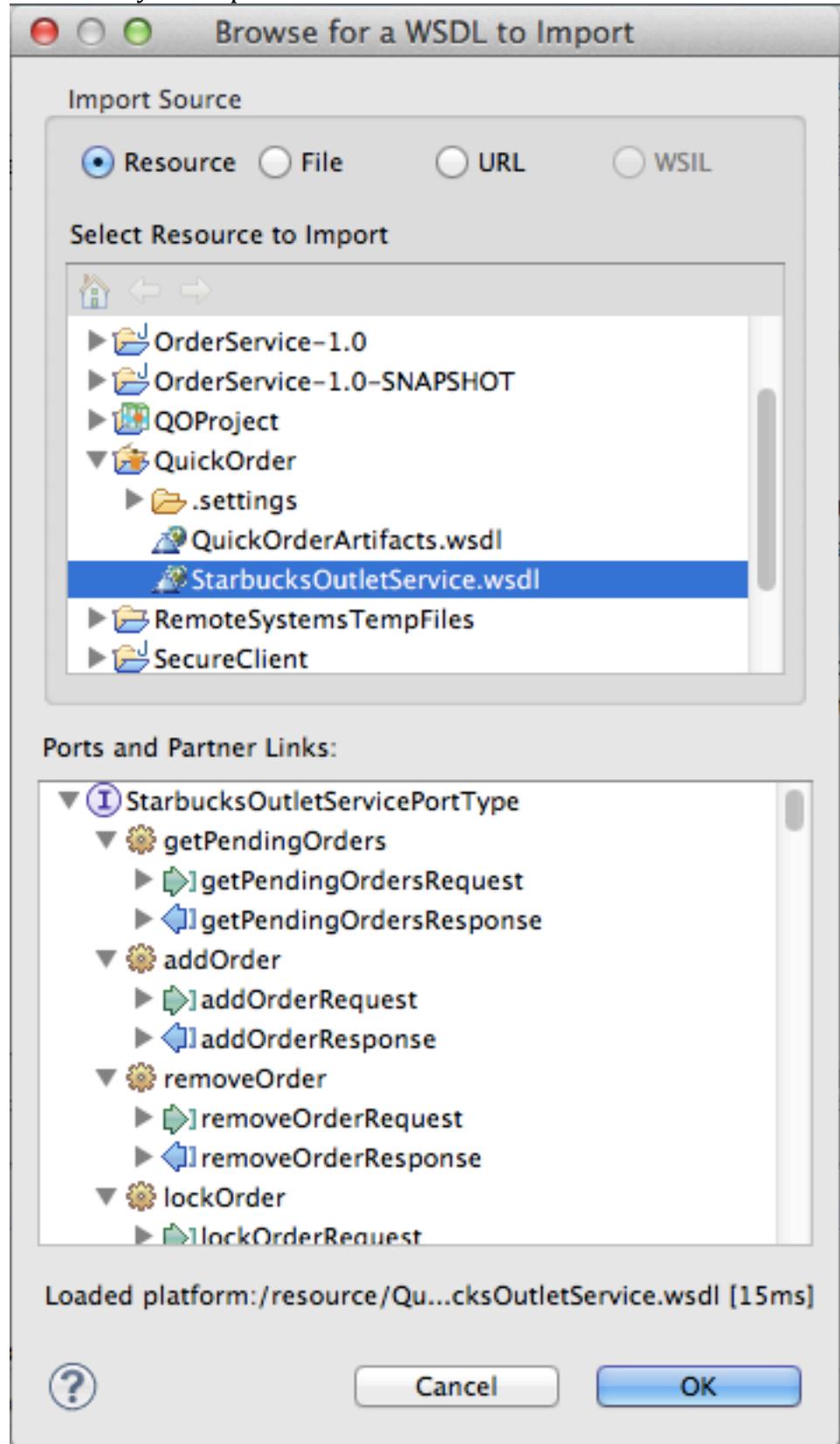
31. In the properties tab below select Details -> Browse:



Click Add WSDL



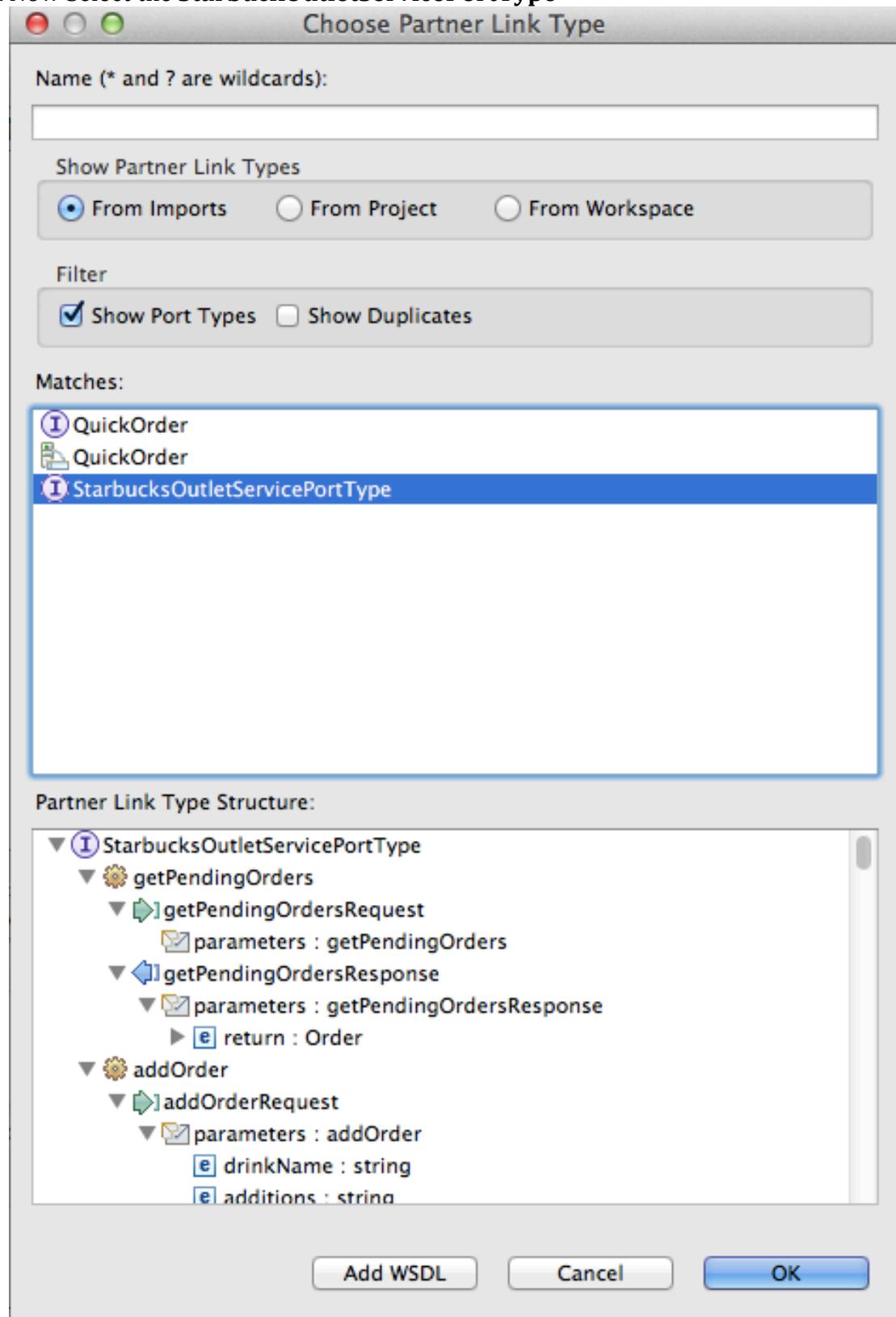
32. Browse to your imported WSDL:



Click **OK**

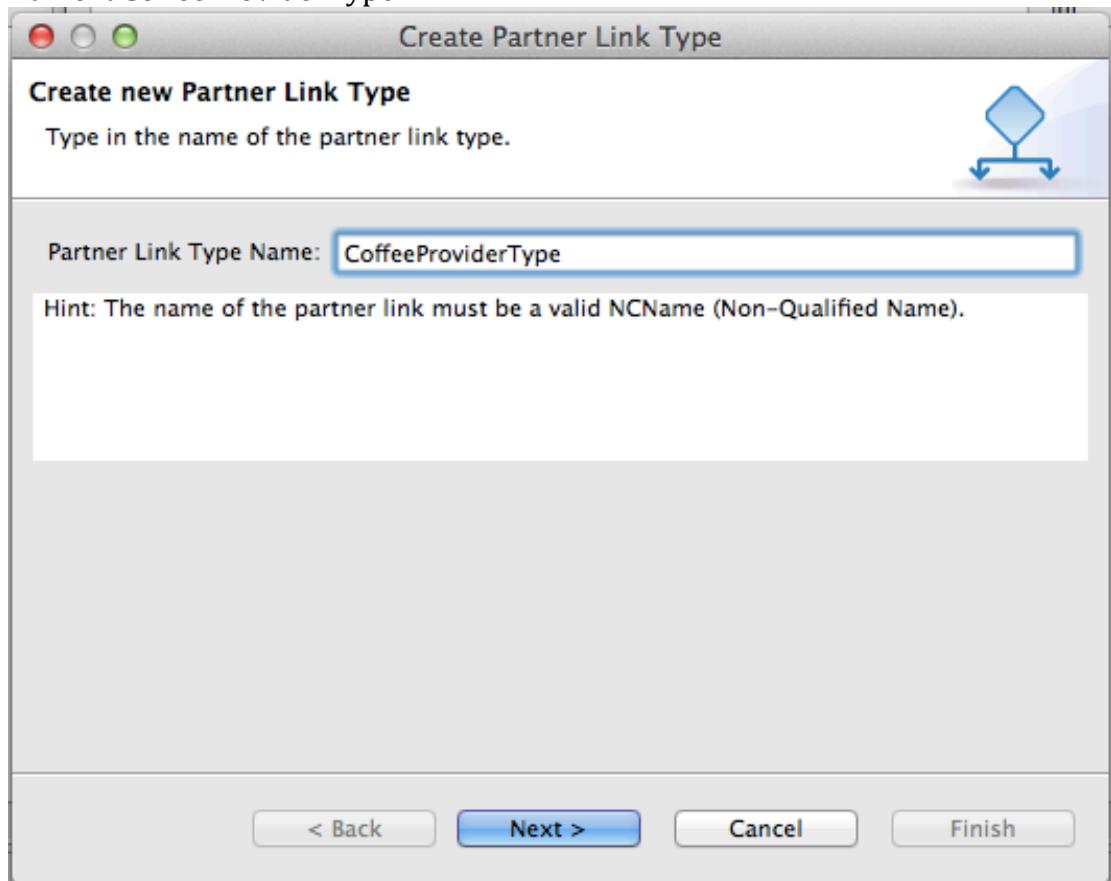


33. Now Select the **StarbuckOutletServicePortType**

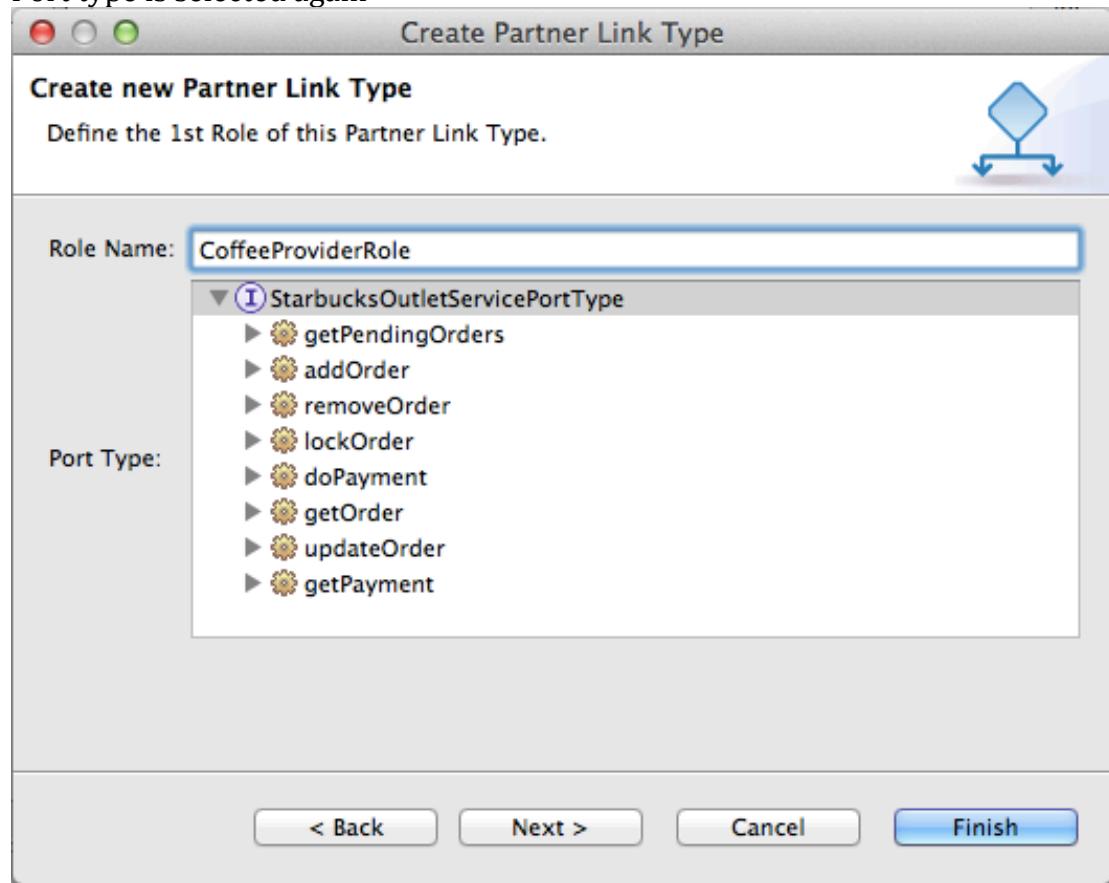


34. Click OK

35. Name it CoffeeProviderType

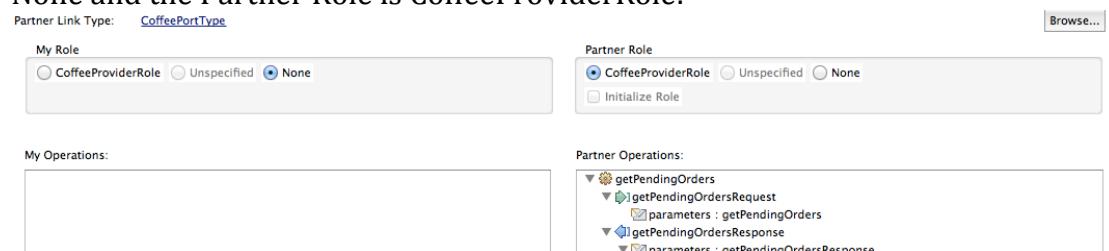


36. Give the role a name (CoffeeProviderRole) and make sure the Starbucks Port type is selected again

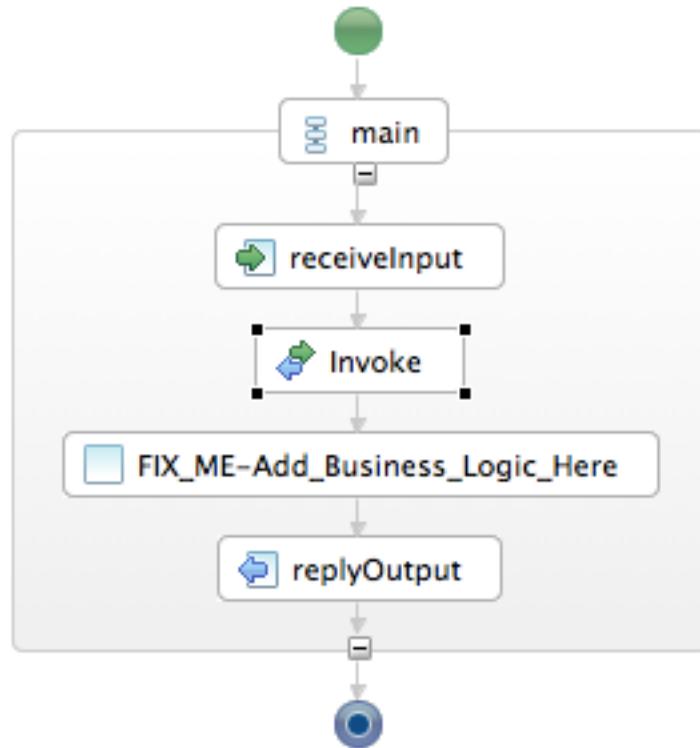


37. Click Finish

38. Back in the Properties pane for the PartnerLink make sure My Role is None and the Partner Role is CoffeeProviderRole:



39. Insert an Invoke before FIX\_ME:



40. Select the invoke. In the properties pane choose the CoffeeProvider partner link and the addOrder operation.

Partner Link: CoffeeProvider

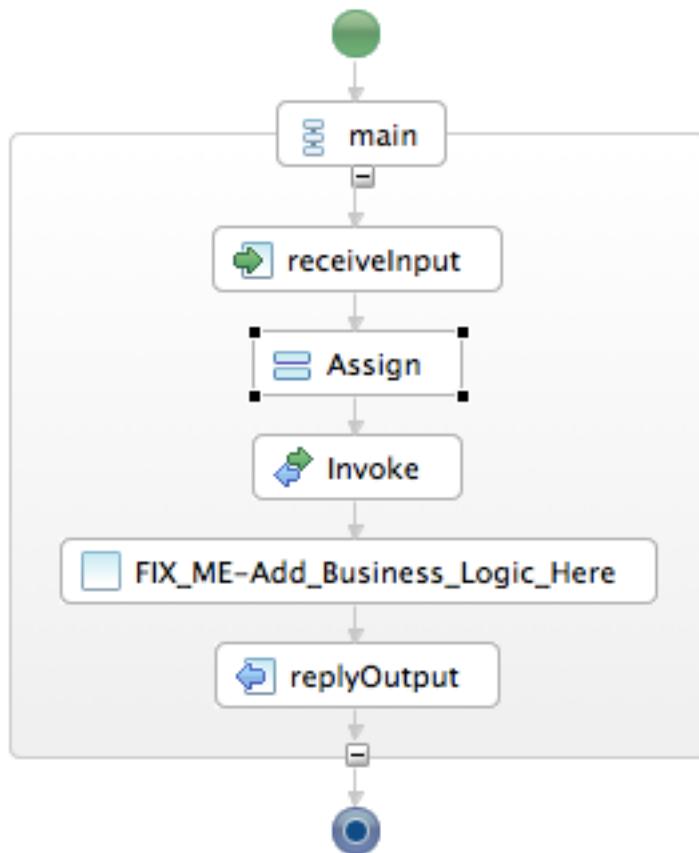
Operation: addOrder

Quick Pick:

- ▼ CoffeeProvider
  - ▼ StarbucksOutletServicePortType
    - ▼ getPendingOrders
      - ▼ getPendingOrdersRequest
        - parameters : getPendingOrders
      - ▼ getPendingOrdersResponse
        - parameters : getPendingOrdersResponse
    - ▼ addOrder
      - ▼ addOrderRequest
        - parameters : addOrder
      - ▼ addOrderResponse
        - parameters : addOrderResponse



41. Now insert an Assign before Invoke:

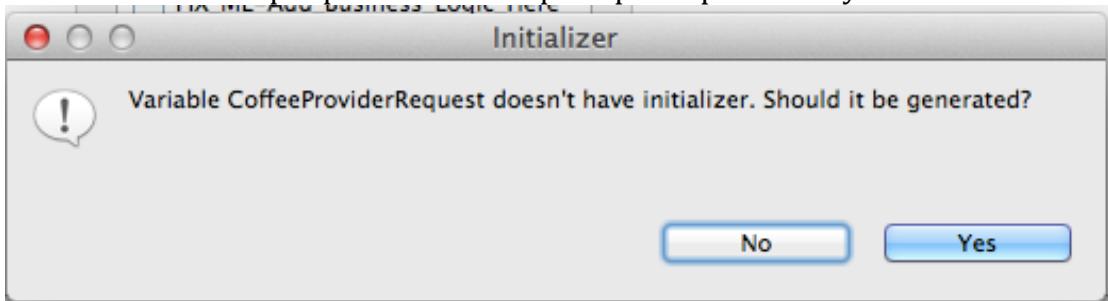


42. In the Properties pane below, click New

43. Choose Variable to Variable

44. Choose the drink from the input/payload and map it to the drinkName in the CoffeeProviderRequest

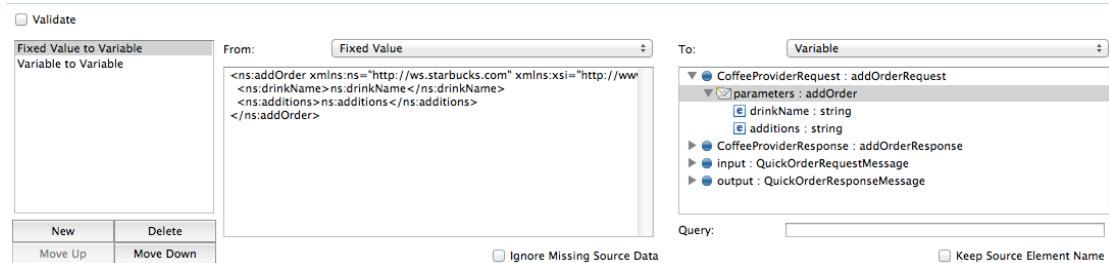
45. Click back on the input pane and it will prompt Eclipse to ask you



Click Yes

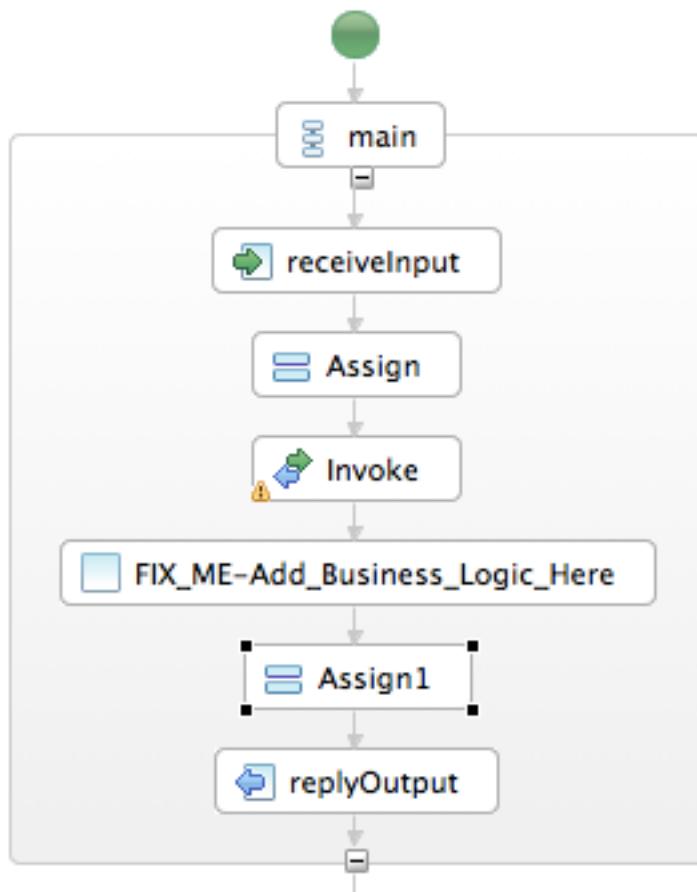
46. This will auto generate a second “copy” operation which is required by the BPEL spec to initialize the XML message for the call out to Starbucks. Your properties should now look like:





Remove the contents of the additions element, so it reads <ns:additions/>

47. Create another Assign before replyOutput



48. Get it to copy from **CoffeeProviderResponse/payload/orderId** into **output/payload/response** [This is several steps equivalent to steps 38-42]

49. Delete the FIXME thingy.

50. Save the BPELProcess.

51. Go to the deploy.xml



52. Choose the right inbound port type for the client partnerlink:  
QuickOrderPort.
53. Choose the StarbucksOutletServiceHttpSoap11Endpoint for the  
CoffeeProvider partnerlink.

Our process isn't finished, but we should be able to run it.

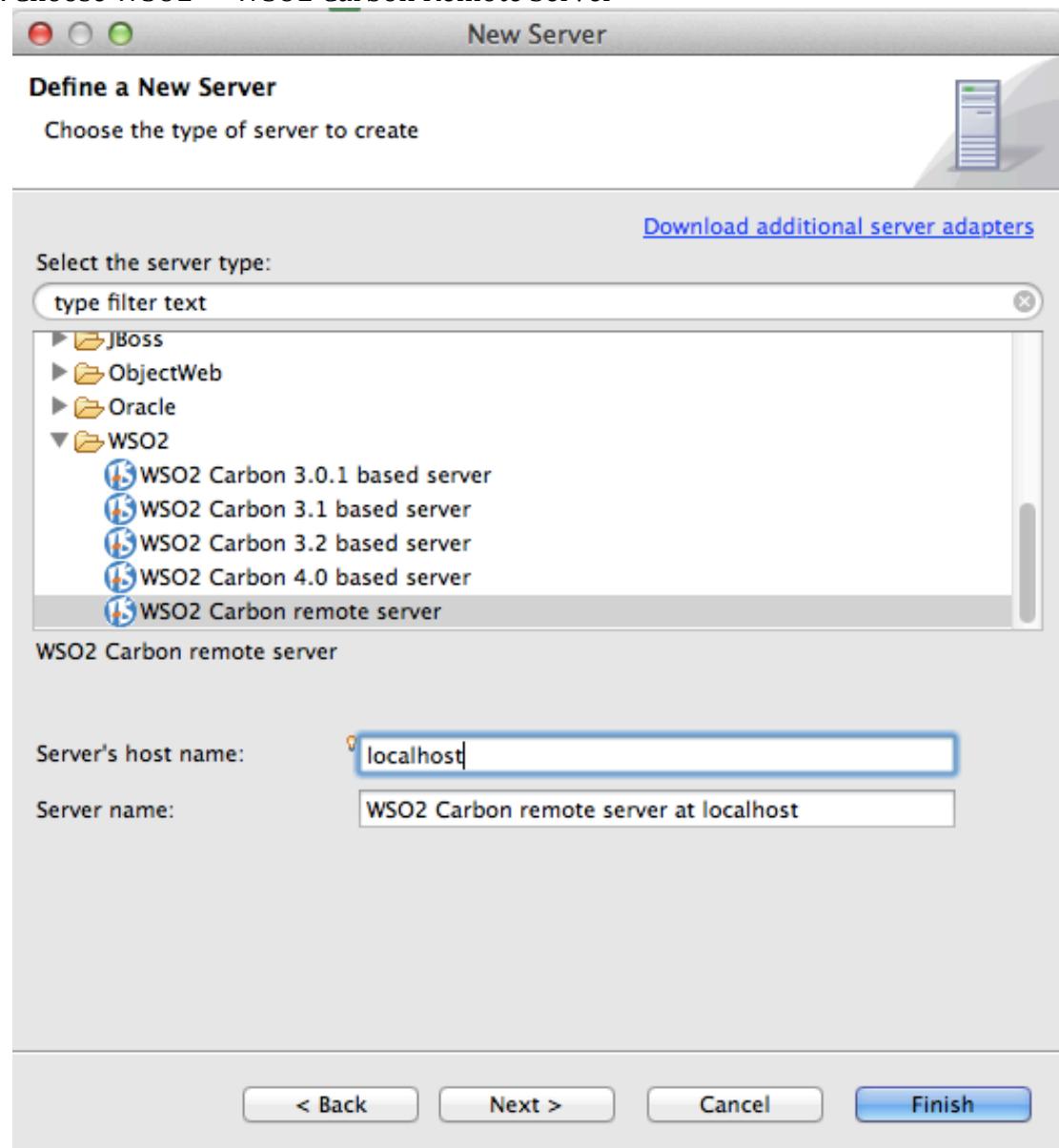
54. Make sure your Business Process Server is running  
`cd ~/servers/wso2bps-3.2.0  
bin/wso2server.sh`

55. Hit Command-N/Ctrl-N to pull up the New dialog.

56. Create a new Server



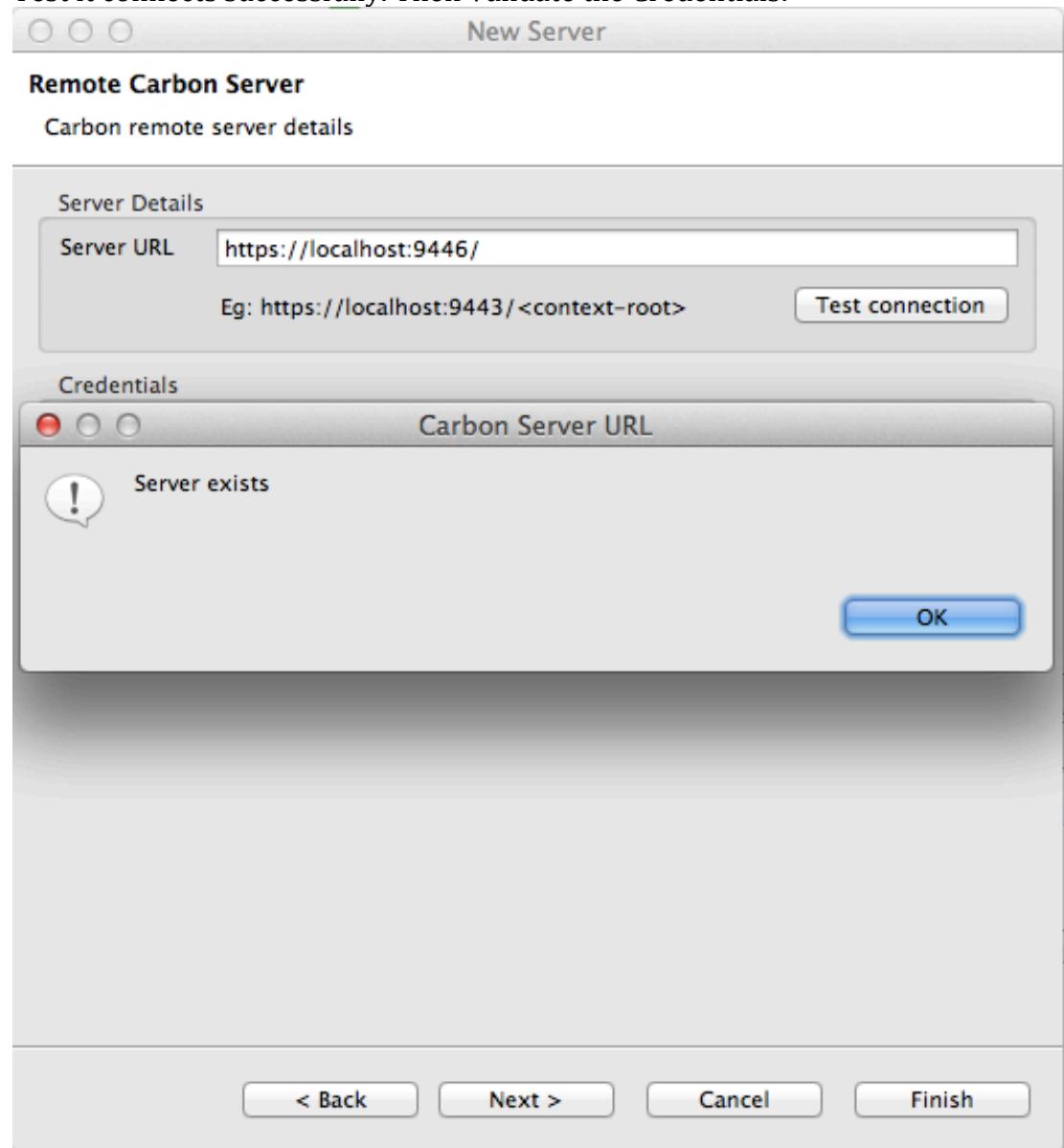
57. Choose WSO2 -> WSO2 Carbon Remote Server



58. Click Next



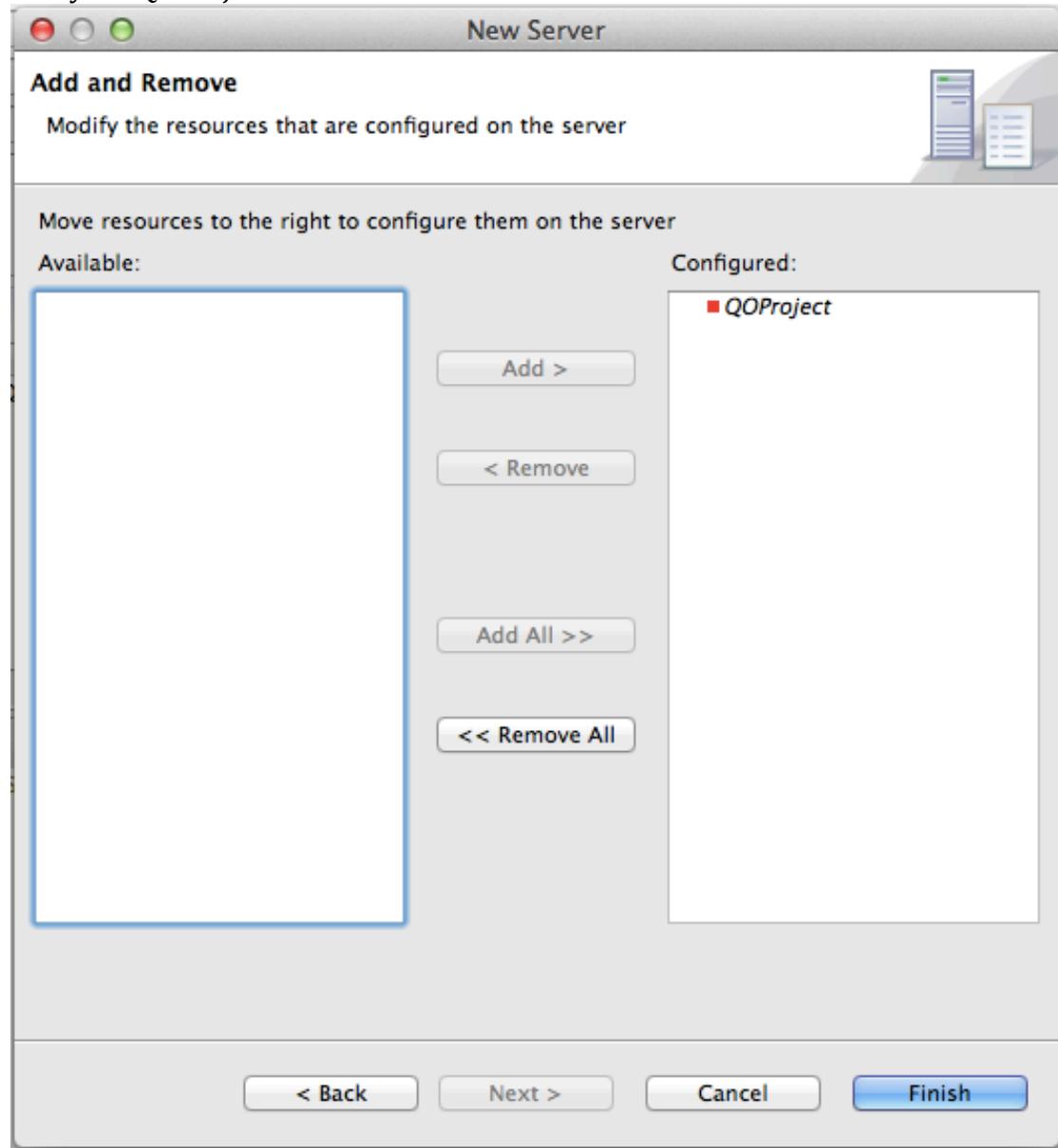
59. Choose the URL of your BPS server (e.g. https://localhost:9446/) Test it connects successfully. Then Validate the Credentials:



60. Click Next



61. Add your QOProject to the Server:



62. Click Finish

63. Go to the QOProject. Right-Click and Run As -> Run on Server



64. Go to your BPS console and wait a bit. Your process should be deployed.

The screenshot shows a table titled "Deployed Processes". It has columns for "Package Name", "Process ID", "Version", "Status", "Deployed Date", and "Manage". There is one row with the following data:

Package Name	Process ID	Version	Status	Deployed Date	Manage
QuickOrder-1.0.0-10	[http://wso2.org/bps/sample]QuickOrder-10	10	ACTIVE	Wed Dec 12 22:34:23 GMT 2012	

65. Click on the QuickOrder-1.0.0-x Process Id. Click Try It.

66. Fill in some plausible data (*make sure your int is an int!*)

The screenshot shows the "Send" interface for a process. It has tabs for "Request" and "Response".

**Request:**

```
<body>
<p:QuickOrderRequest xmlns:p="http://wso2.org/bps/sample">
<!-- Exactly 1 occurrence-->
<name xmlns="http://wso2.org/bps/sample">Paul</name>
<!-- Exactly 1 occurrence-->
<drink xmlns="http://wso2.org/bps/sample">Cappuccino</drink>
<!-- Exactly 1 occurrence-->
<number xmlns="http://wso2.org/bps/sample">4</number>
</p:QuickOrderRequest>
</body>
```

**Response:**

```
<QuickOrderResponse xmlns="http://wso2.org/bps/sample">
<tns:result xmlns:tns="http://wso2.org/bps/sample">66c6c377-f42c-4533-a89d-...
</QuickOrderResponse>
```

67. Hopefully you have created an Order!

68. Ideally you will now do more. The idea is to automate the Ordering and Payment, using a fixed credit card. See if you can get the Process to Order and Pay for a Drink.

69. If you really want to stretch, now get it to Order and pay for n drinks!

