Lab 6

bpm_processor

Lab Goals

- Introduction to bpm_processor in SOA 5
- Introduction to bpm_processor in SOA 6
- Step-by-step migration using Windup rules
- Deploy and test application in SOA 6

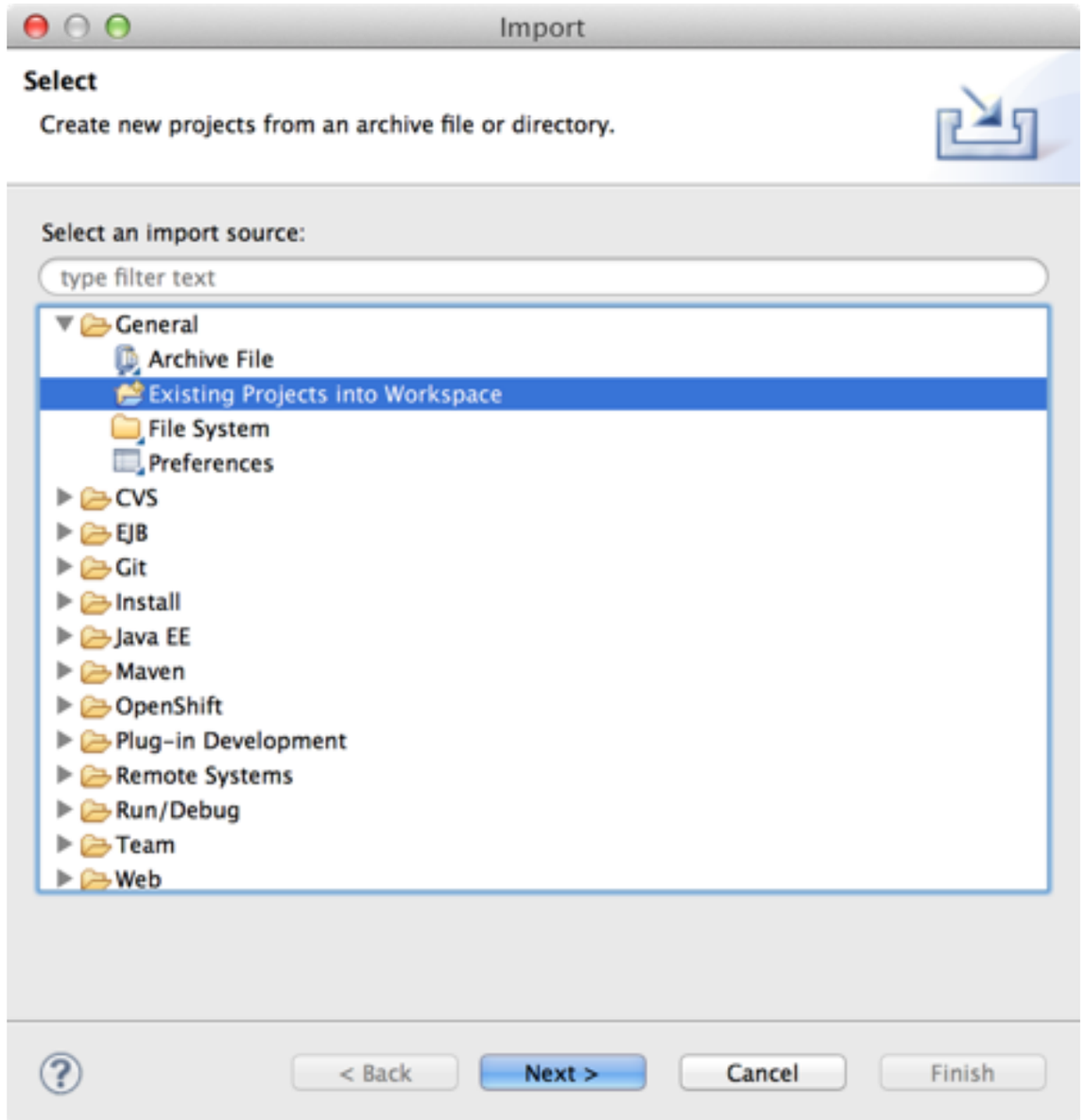
Importing SOA 5 BPM Processor

TODO

1. File -> Import ... from the JBDS menu.

2. Select General -> Existing Projects into Workspace

3. Click Next



Importing SOA 5 BPM Processor

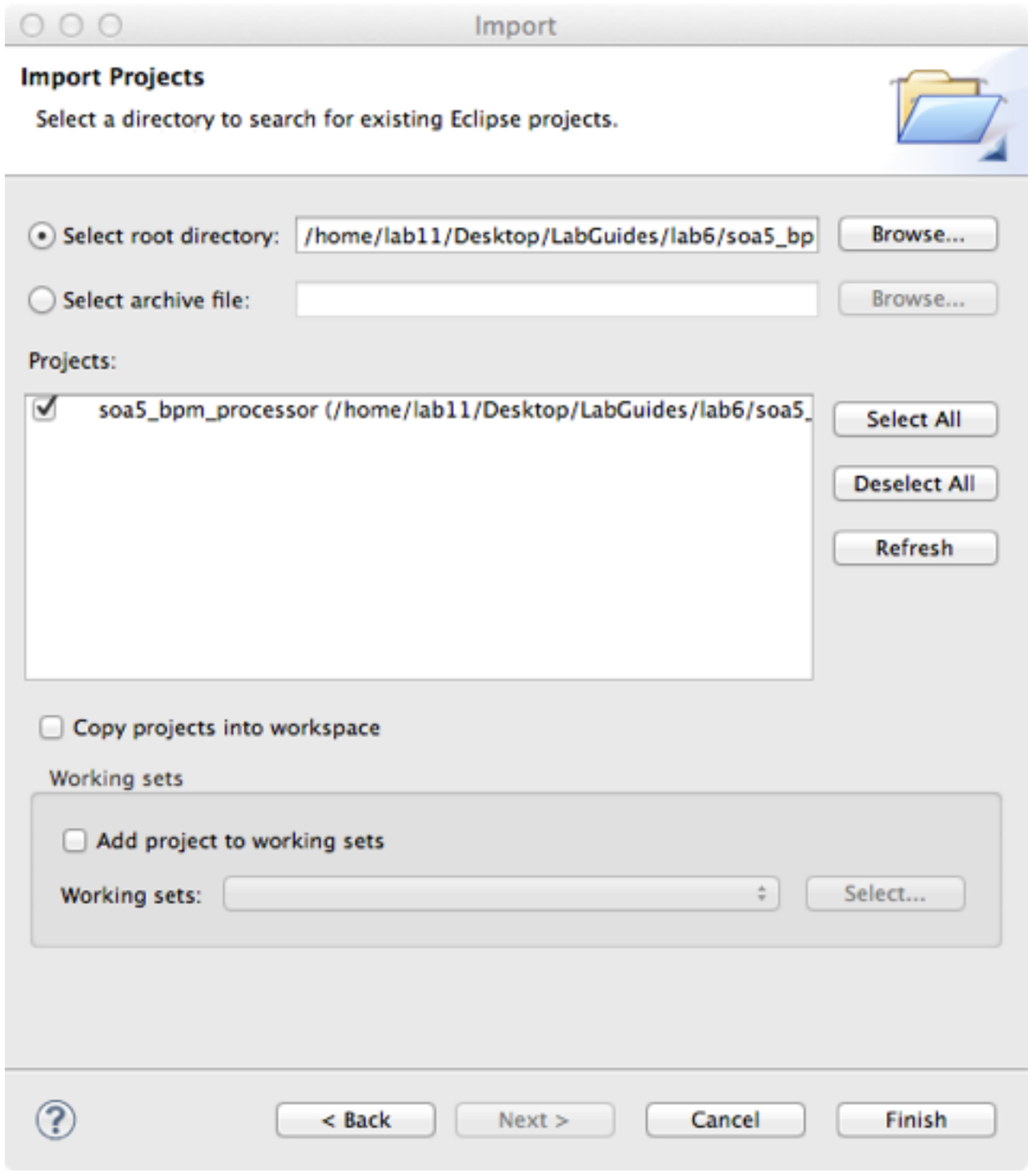
TODO

1. Click Browse ... and navigate to:

`/home/lab11/Desktop/LabGuides/lab6/soa5_bpm_processor`

2. Make sure the soa5_bpm_processor project is checked

3. Click Finish



Files To Note

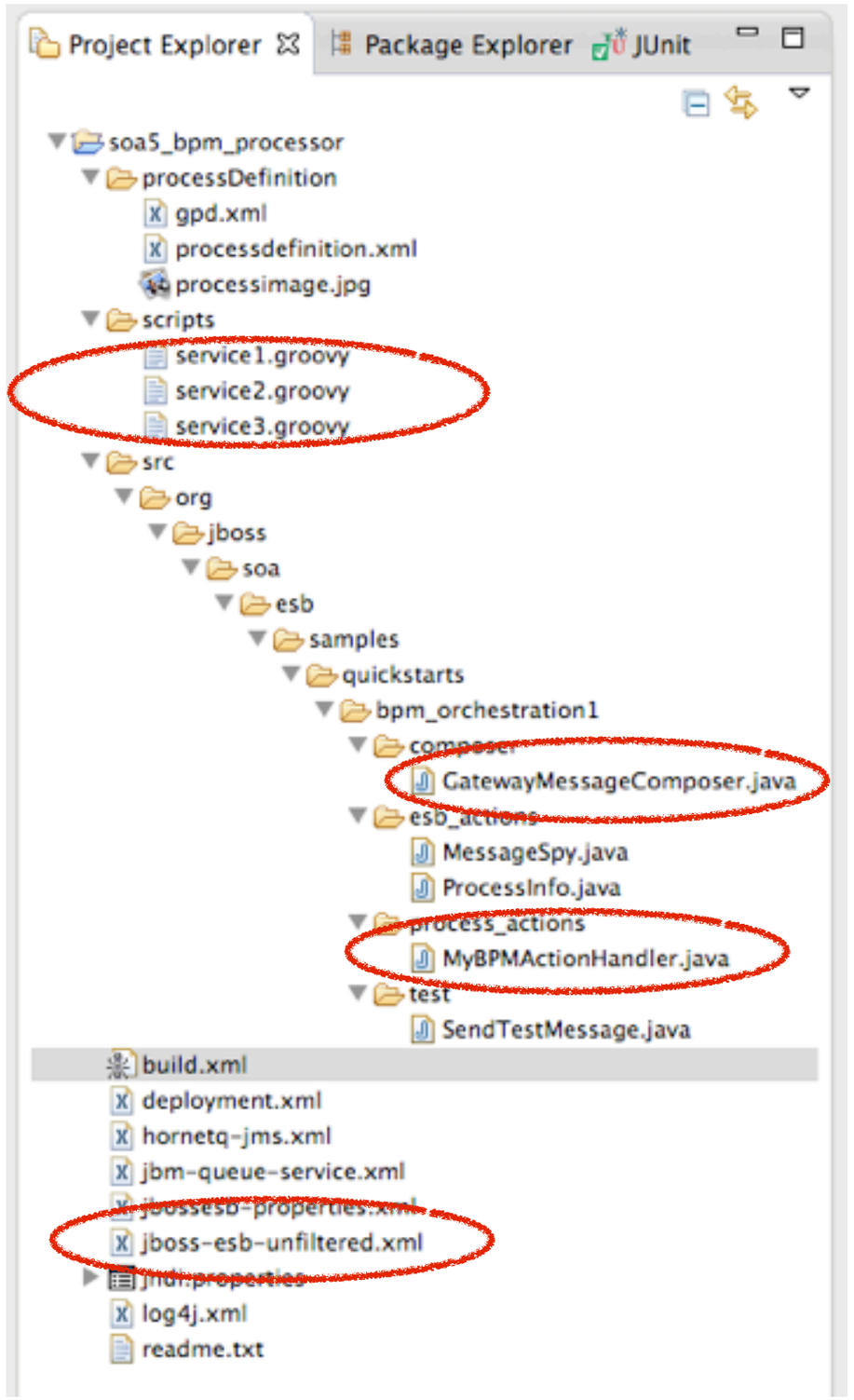
TODO

1. `jboss-esb-unfiltered.xml` contains an unfiltered version of the project's service definitions and configuration. Open the file by double-clicking on it in the Project Explorer.

2. The `scripts` directory contains three groovy scripts which are used to modify the message.

3. `MyBPMActionHandler.java` contains logic which will be executed by the BPM process.

4. `GatewayMessageComposer.java` is a custom message composer.



jboss-esb-unfiltered.xml

jboss-esb-unfiltered.xml

```
<?xml version = "1.0" encoding = "UTF-8"?>
<jbossesb xmlns="http://anonsvn.labs.jboss.com/labs/jbossesb/trunk/product/etc/schemas/xml/jbossesb-1.0.1.xsd"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://anonsvn.labs.jboss.com/labs/jbossesb/trunk/product/etc/schemas/xml/jbossesb-1.0
    parameterReloadSecs="5">

    <providers>
        <fs-provider name="FSprovider1">
           <fs-bus busid="startGwChannel" >
               <fs-message-filter
                   directory="@INPUT_DIR@"
                   input-suffix=".startProcessGW"
                   work-suffix=".startProcessGWWorking"
                   post-delete="true"
                   error-delete="true"
               />
           </fs-bus>
           <fs-bus busid="startEsbChannel" >
               <fs-message-filter
                   directory="@INPUT_DIR@"
                   input-suffix=".startProcessESB"
                   post-delete="true"
                   error-delete="true"
               />
           </fs-bus>
        </fs-provider>
        <jms-provider name="JMSProvider" connection-factory="ConnectionFactory">
           <jms-bus busid="service1EsbChannel" >
              <jms-message-filter dest-type="QUEUE"
                dest-name="queue/quickstart_bpm_orchestration1_service1_Request_esb"/>
           </jms-bus>
           <jms-bus busid="service2EsbChannel" >
              <jms-message-filter dest-type="QUEUE"
                dest-name="queue/quickstart_bpm_orchestration1_service2_Request_esb"/>
           </jms-bus>
           <jms-bus busid="service3EsbChannel" >
              <jms-message-filter dest-type="QUEUE"
```

Design Source

service1.groovy

service1.groovy

```
import org.jboss.soa.esb.message.*

println "1********** Begin Service 1 ************"

println "In: " + new String(message.getBody().get())

message.getBody().add("Service 1 " + new String(message.getBody().get()))

println "Out: " + message.getBody().get()

println "************* End Service 1 *************"
```

MyBPMActionHandler.java

MyBPMActionHandler.java

```
 * JBoss, Home of Professional Open Source
package org.jboss.soa.esb.samples.quickstarts.bpm_orchestration1.process_actions;

import org.jbpm.graph.def.ActionHandler;

public class MyBPMActionHandler implements ActionHandler
{
    private static final long serialVersionUID = 1L;

    Long startFrom;

    public void execute(ExecutionContext exCtx) throws Exception
    {
        // Token token = exCtx.getToken();
        // ContextInstance context = token.getProcessInstance().getContextInstance();
        // Object obj   = context.getVariable("counter",token);

        System.out.println("Executed by the process, not by the ESB");

        //context.setVariable("counter", counter , token);
    }

}
```

GatewayMessageComposer.java

GatewayMessageComposer.java

```
 * JBoss, Home of Professional Open Source
package org.jboss.soa.esb.samples.quickstarts.bpm_orchestration1.composer;

import java.io.File;

public class GatewayMessageComposer extends LocalFileMessageComposer<File>
{
    private static final String DEFAULT_CHARSET = "UTF-8" ;

    private String charset = DEFAULT_CHARSET ;

    public void setConfiguration(final ConfigTree config)
        throws ConfigurationException
    {
        super.setConfiguration(config) ;
        charset = config.getAttribute("encoding", DEFAULT_CHARSET) ;
    }
    protected Object getPayload(final File inputFile)
        throws IOException, MimeDecodeException
    {
        final Object contents = super.getPayload(inputFile) ;
        if (contents instanceof byte[])
        {
            final byte[] bytes = (byte[]) contents ;
            return new String(bytes, charset) ;
        }
        else
        {
            return contents ;
        }
    }

    public Object decompose(final Message message, final File inputFile)
        throws MessageDeliverException
    {
        final Object payload = super.decompose(message, inputFile) ;
        if (payload instanceof String)
        {
            final String result = (String)payload ;
```

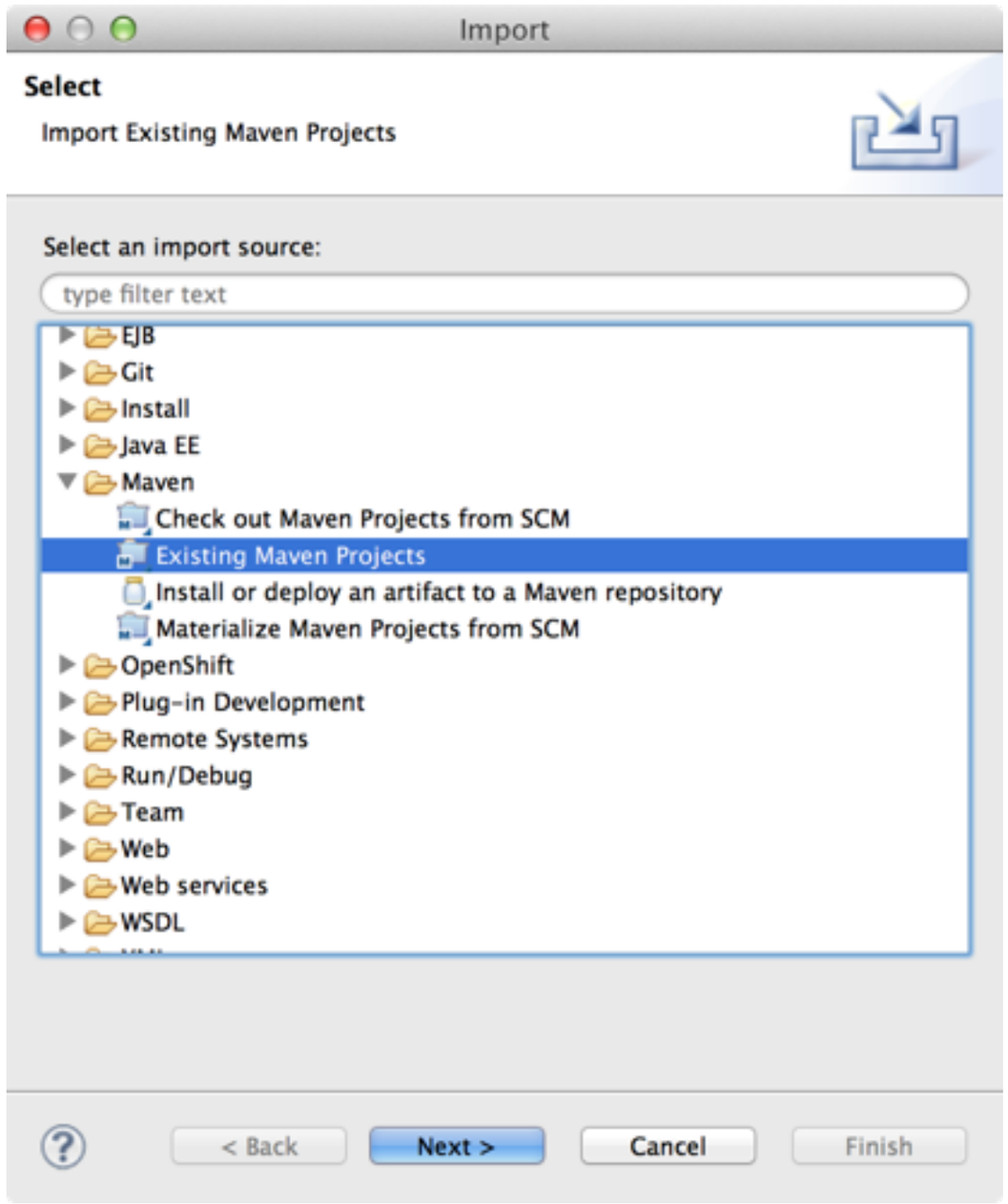
Importing SOA 6 BPM Processor

TODO

1. File -> Import ... from the JBDS menu.

2. Select Maven -> Existing Maven Projects

3. Click Next



Importing SOA 6 Hello World

TODO

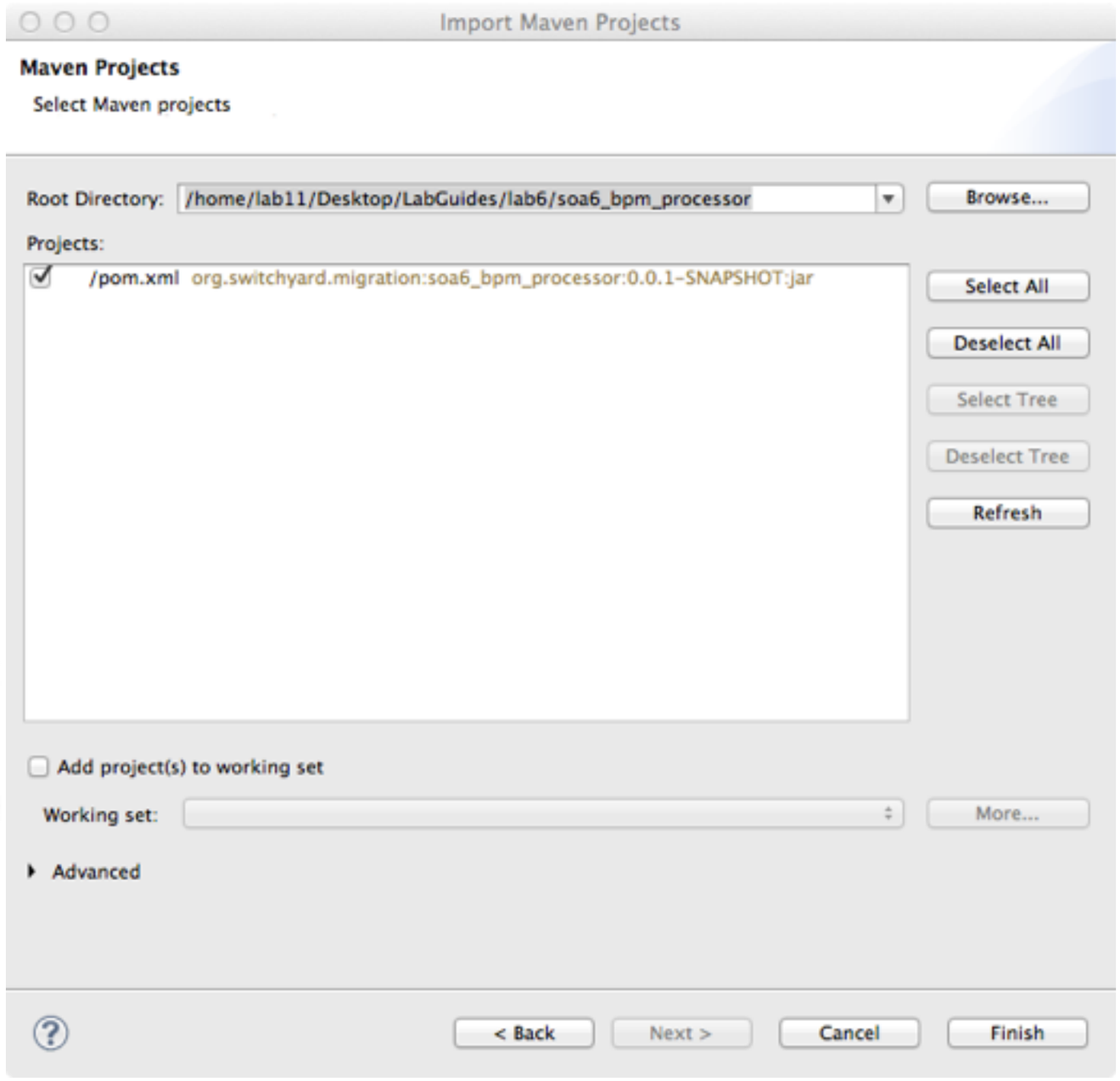
1. Click Browse ... and navigate to:

`/home/lab11/Desktop/LabGuides/lab6/soa6_bpm_processor`

2. Make sure the pom.xml is checked for:

`org.switchyard.migration:soa6_bpm_processor`

3. Click Finish



Get Your Windup On!

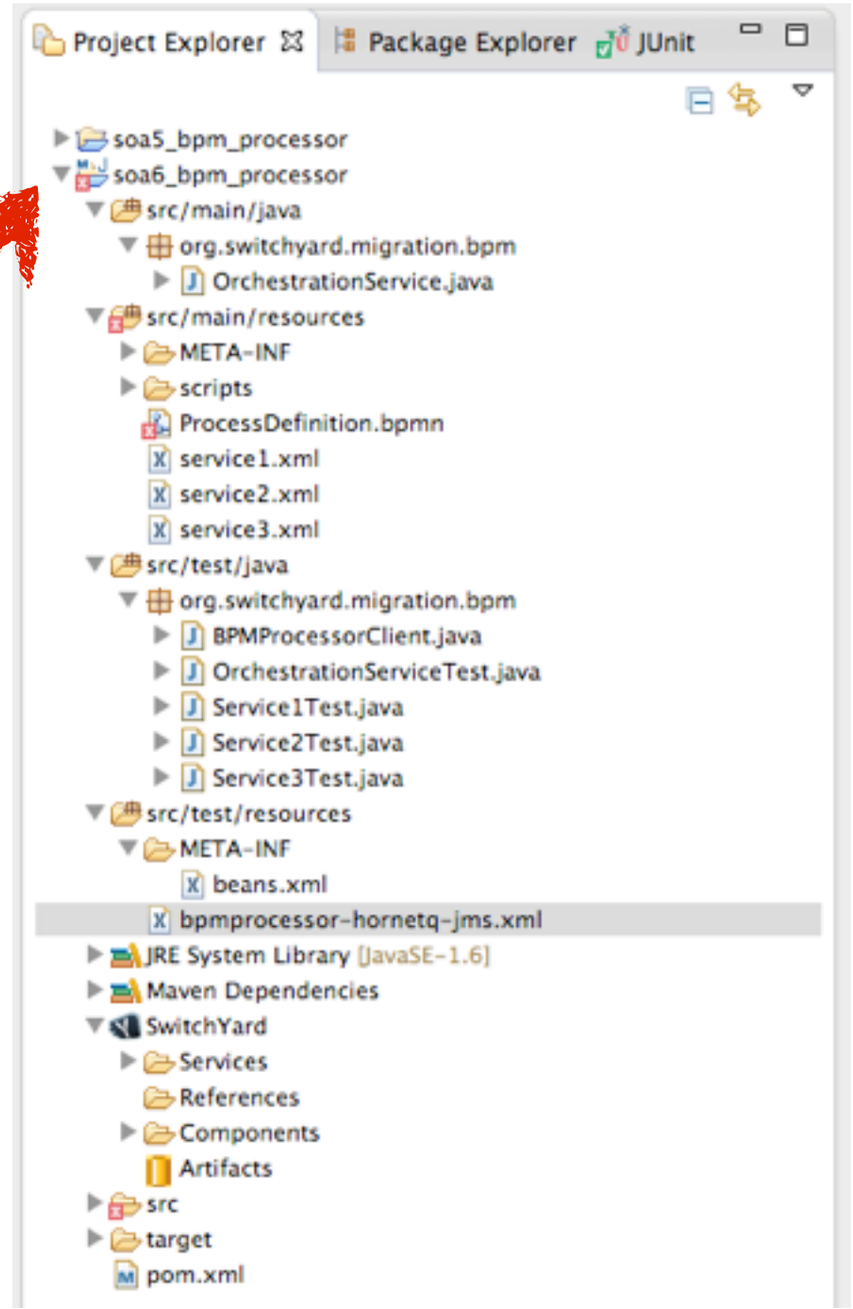
FYI

OK, you know the drill at this point. Check out the windup report and check out the SOA 5 and SOA 6 apps to see the changes.

Deploy Application

TODO

1. Right-click on the soa6_bpm_processor project and select Run As ... Run on Server

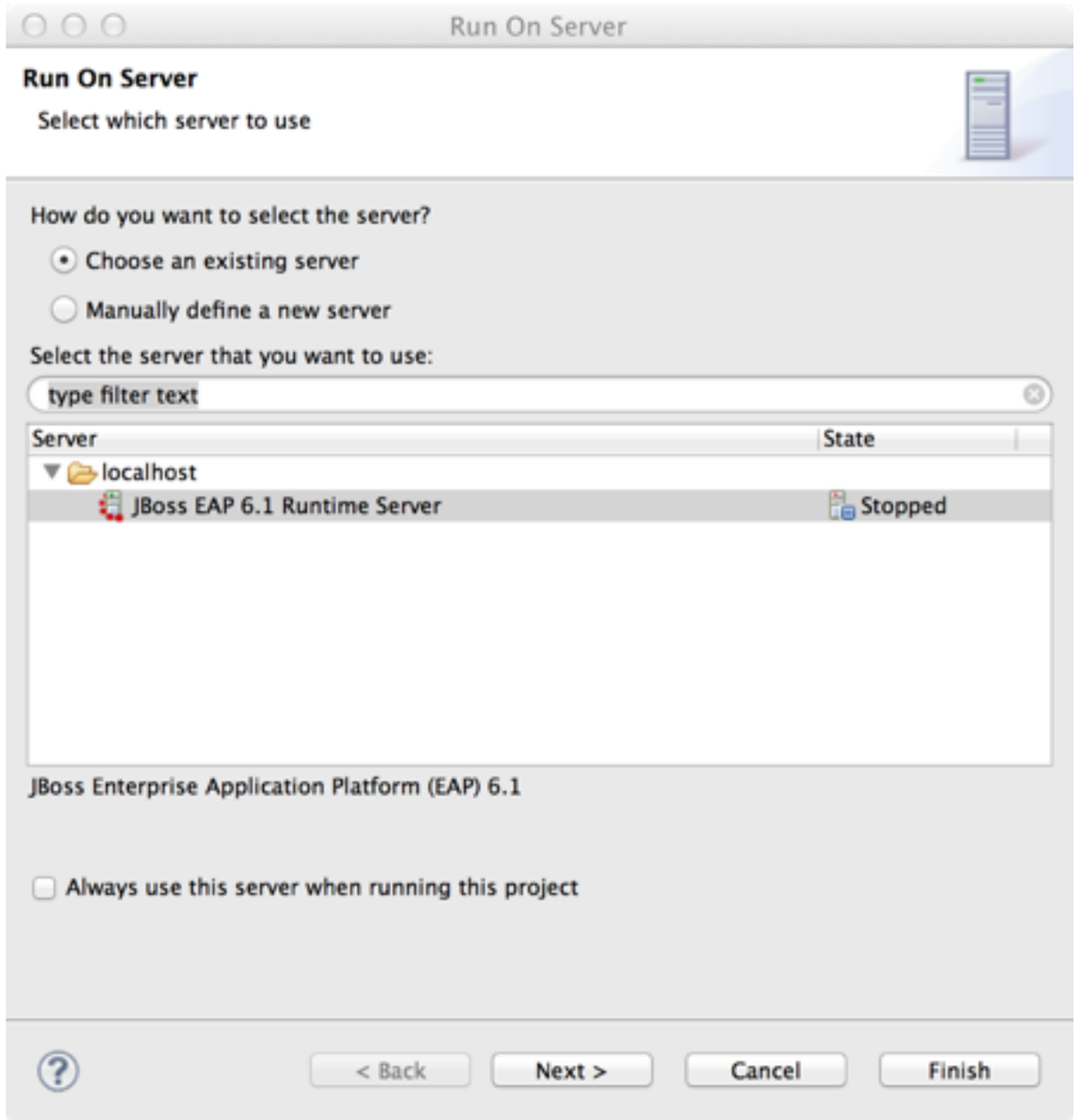


Select Server

TODO

1. Select the JBoss EAP 6.1 Runtime Server

2. Click Finish

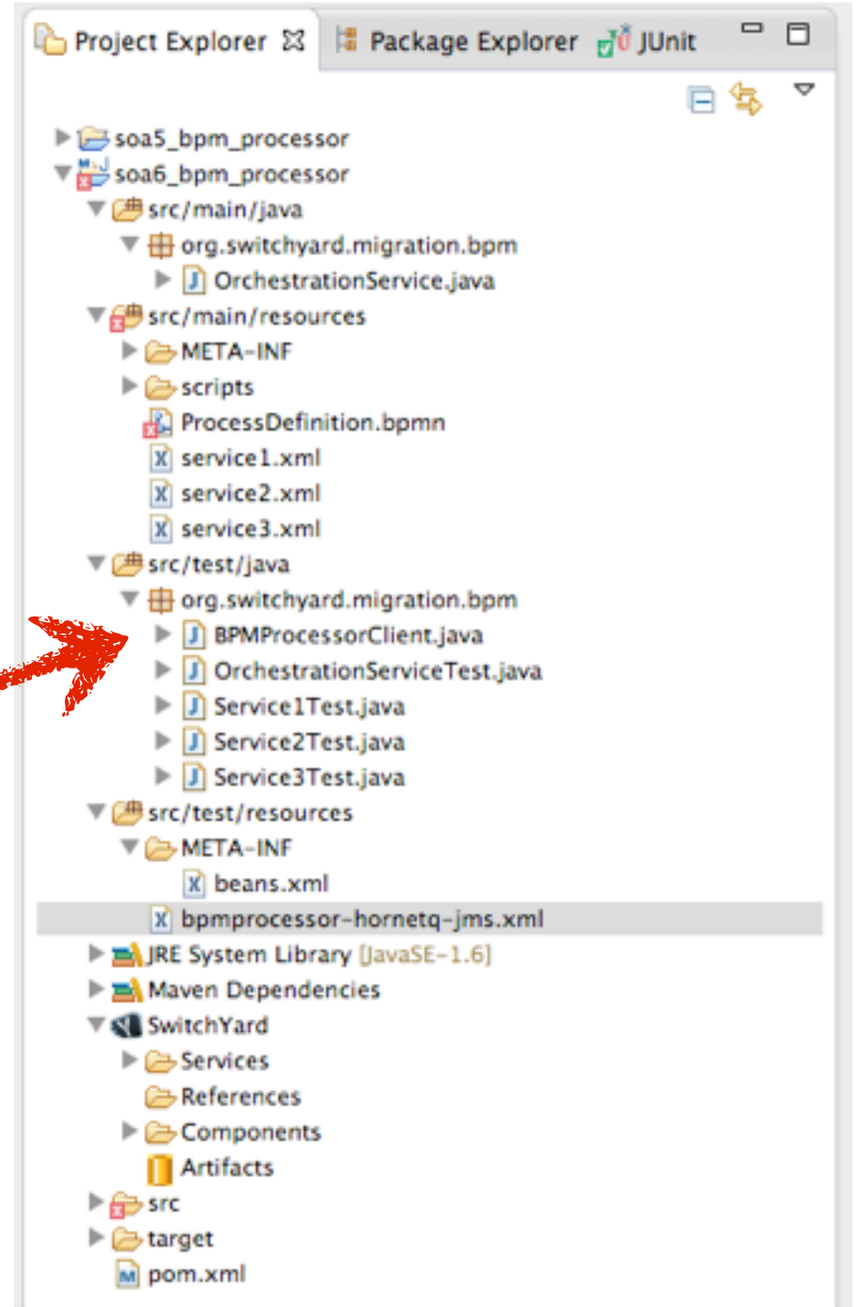


Run Test Client

TODO

1. Open HelloWorldClient.java from the Project Explorer view.

2. Go to the Run menu in the main menu bar and select ‘Run As -> Java Application’



Verify Output

TODO

1. Click here to swap between application output and server output.

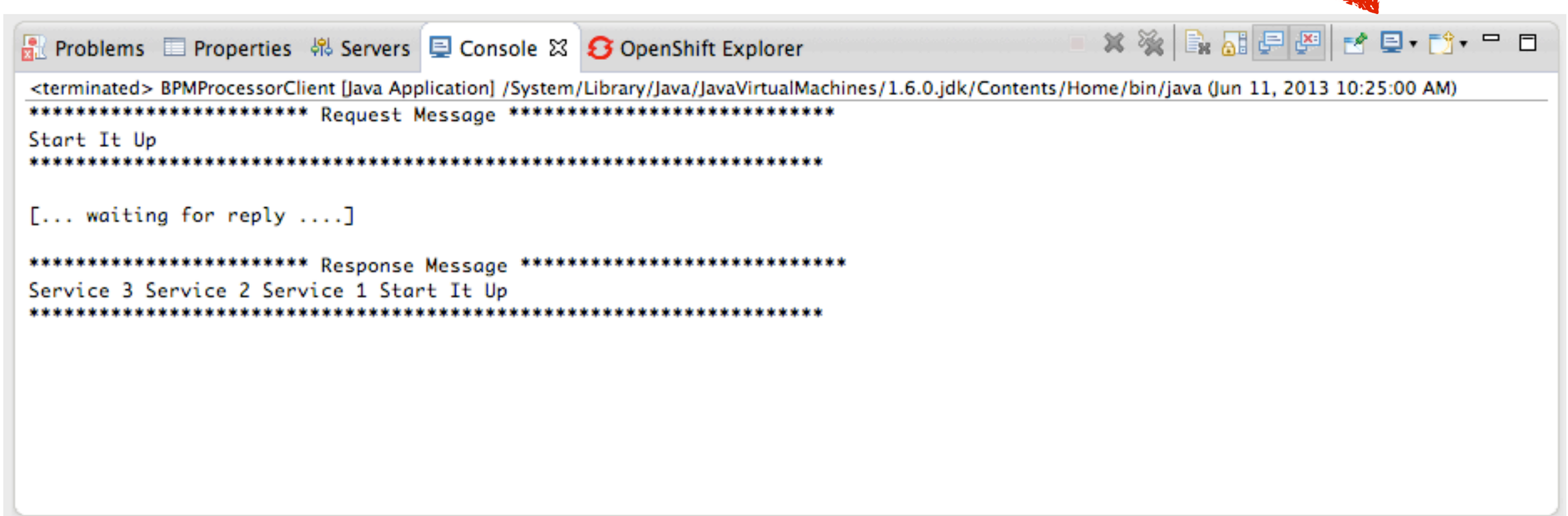
Problems | Properties | Servers | Console | OpenShift Explorer

```
JBoss 7.1 Runtime 1 Server [JBoss Application Server Startup Configuration] /System/Library/Java/JavaVirtualMachines/1.6.0.jdk/Contents/Home/bin/java (Jun 11, 2013 1
Service -> {urn:org.switchyard.migration:soa6_bpm_processor:1.0}OrchestrationService
Operation -> process
Phase -> OUT
State -> OK
Exchange Context ->
        org.switchyard.exchange.transaction.beforeInvoked : true
        org.switchyard.exchangeDurationMS : 1018
Message Context ->
        org.switchyard.relatesTo : 4c17af0a-2a8b-4ea3-93f9-2cb40c024648
        org.switchyard.contentType : java:java.lang.String
        {urn:switchyard-component-bpm:bpm:1.0}processInstanceId : 1
        org.switchyard.messageId : fab94a7a-47ea-4d96-8586-d99e3ea8b27f
Message Content ->
Service 3 Service 2 Service 1 Start It Up
------ End Message Trace -------
```

Verify Output

TODO

1. Click here to swap between application output and server output.



Lab 6 Complete!