ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

Εισαγωγή στα Δίκτυα Υπηρεσιών

Assisting Lecture 11 - Apache ODE,Eclipse BPEL Designer Examples

Μύρων Παπαδάκης
Τμήμα Επιστήμης Υπολογιστών

Introduction to Service Networks
CS-592 – Spring 2015

Assisting Lecture : Eclipse BPEL Designer

Myron Papadakis (myrpap@gmail.com)

Outline

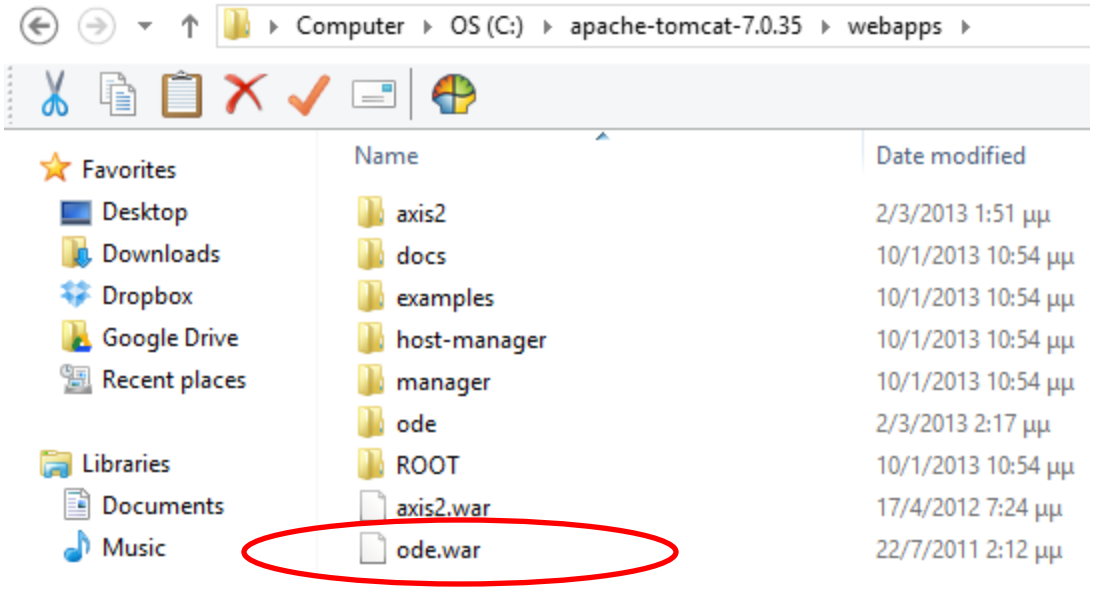
- Apache Ode
- Eclipse BPEL Designer

Apache Ode

- Apache ODE (Orchestration Director Engine) is an open source BPEL engine.
- Full support for WS- BPEL 2.0 OASIS standard.
- Hot-deployment of processes.
- Management interface for processes, instances and messages.

Apache Ode

- Download latest stable release: ODE 1.3.5 war distribution from
  http://ode.apache.org/getting-ode.html
- Installation:
  - Unzip package. Copy this file to Tomcat's **webapp**.
  - Start Tomcat and ODE should be up and running.



Testing Apache Ode

http://localhost:8080/ode

Deploying a process to Apache Ode

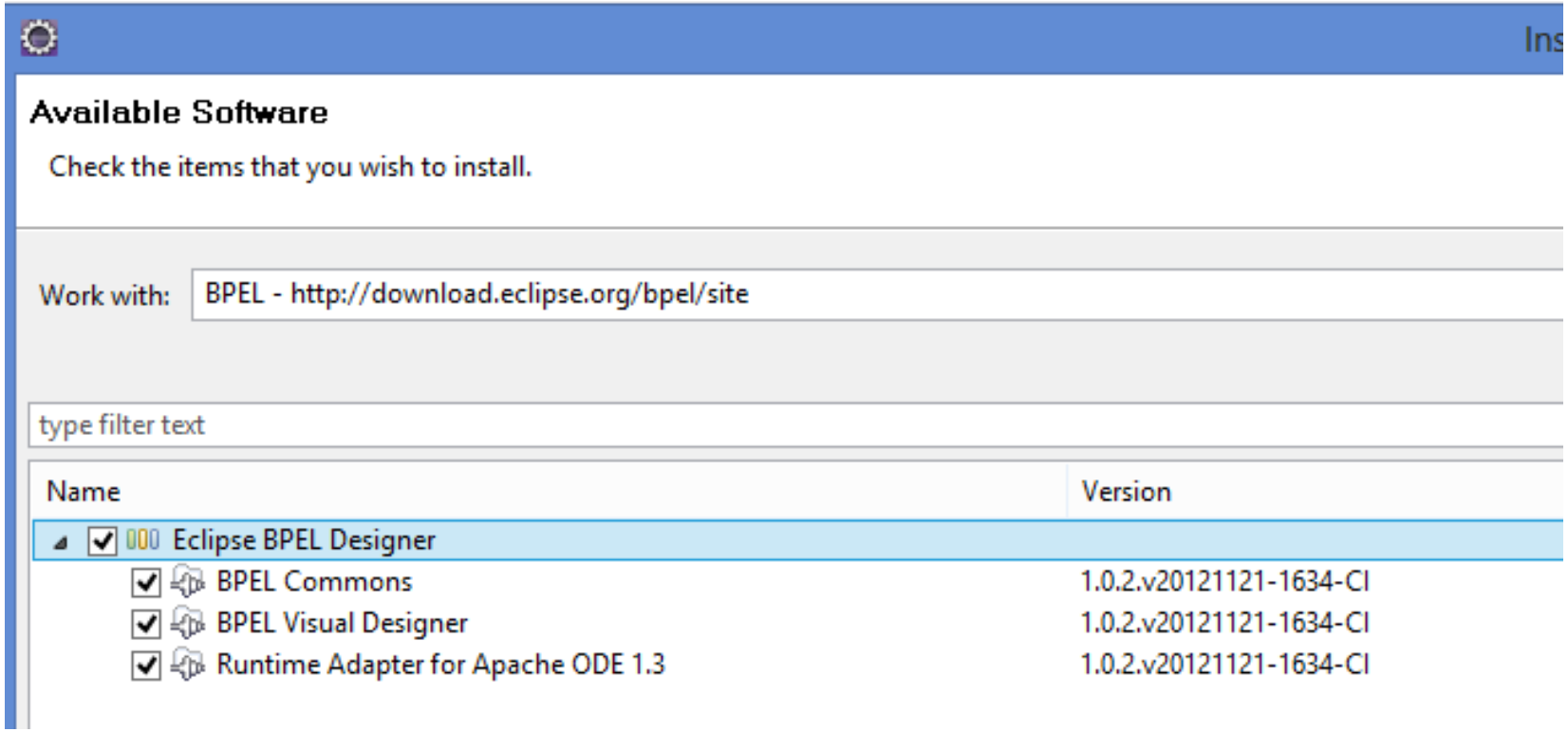
- http://ode.apache.org/creating-a-process.html
- Each deployment is a directory with a relevant deployment artifacts
  - One or more processes (.bpel)
  - WSDL and XSD files
  - *Deployment descriptor* (***deploy.xml***): configuration of services that processes uses. For each process,
    - partnerlinks are bind to concrete WSDL services
    - <receive> is correlated to <provide> element
    - <invoke> is correlated to <invoke> element
- To deploy, copy the **whole** directory containing your artifacts in the
  - "***<tomcat-home>/webapps/ode/WEB-INF/processes***"

Eclipse

Eclipse and BPEL

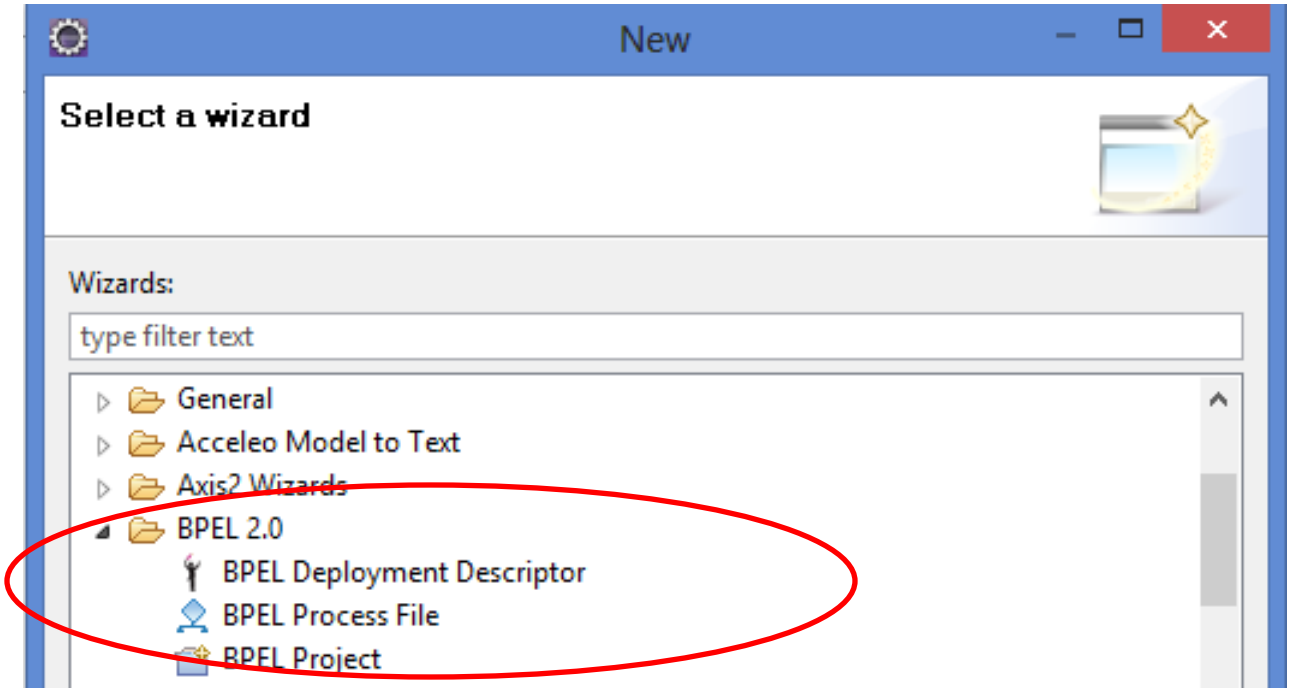
- Installing and Configuring Eclipse with the BPEL Designer
- For installing plug-ins and additional packages go to: **Help → Install New Software**
- In the text field with label "Work With" enter http://download.eclipse.org/bpel/site and from the list select "**Eclipse BPEL Designer**"
- After installation of new packages, **Eclipse IDE must be restarted**.
- In order to check installed software, plug-ins and tools, try: ***"Help"*** ***→"About Eclipse"*** and click ***"Installation Details"***

Eclipse BPEL Designer



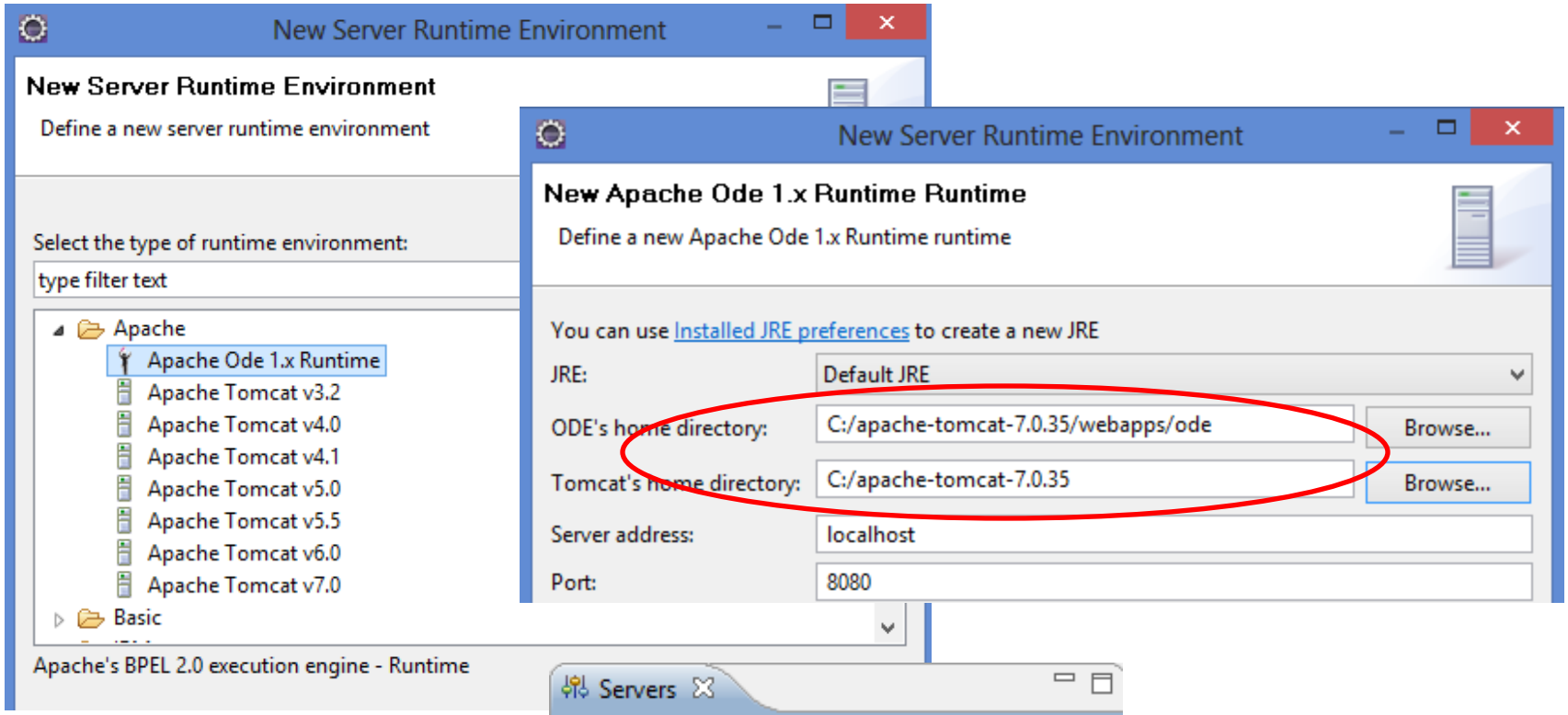
Eclipse and BPEL

- To verify the installation has been successful, after the restart, click on File → New → Other (in order to create a new project)
- Verify that there is a BPEL 2.0 Entry in the Wizard



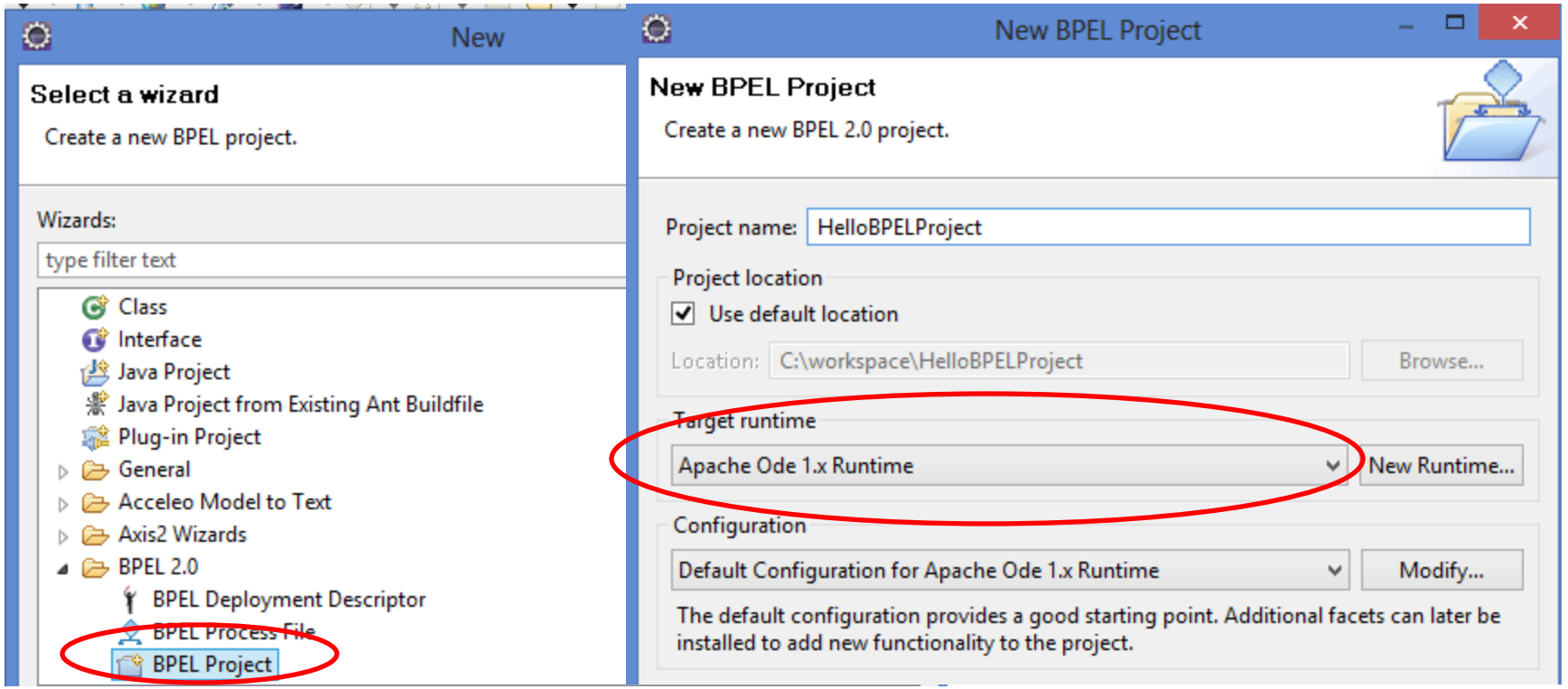
Eclipse and Apache Ode

- Setup a ODE Server inside Eclipse
  - "Window→***Preferences->Server->Server Runtime***" and "***Add***"

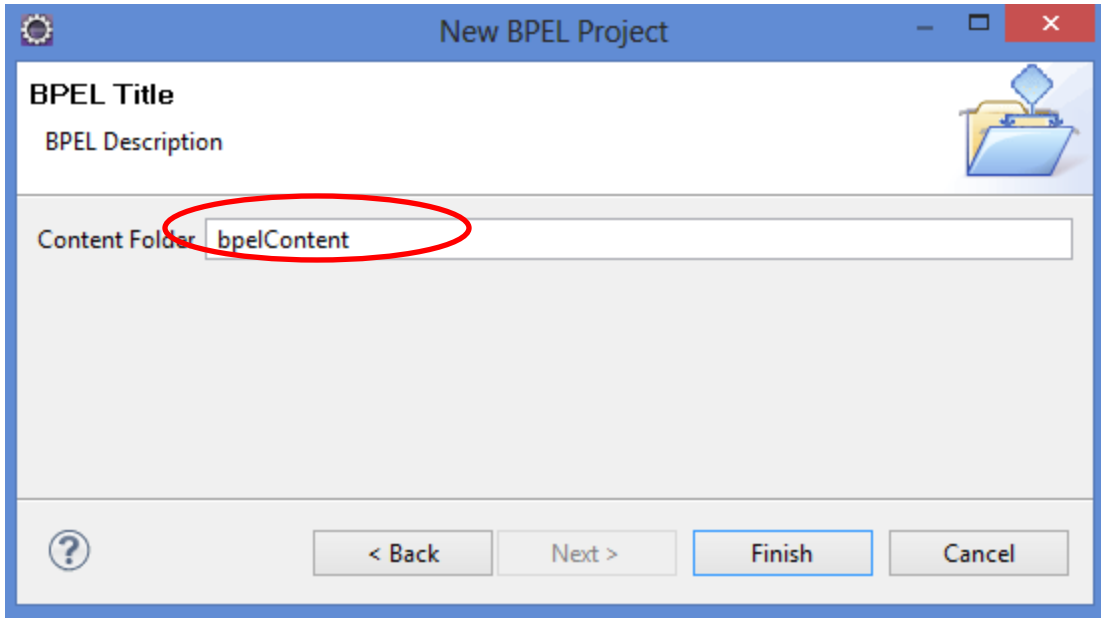


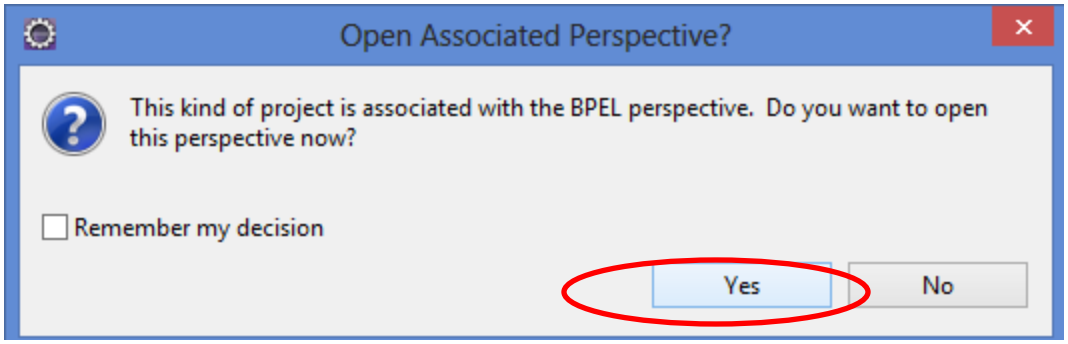
Eclipse BPEL Project

- File → New → Other → BPEL 2.0 → BPEL Project



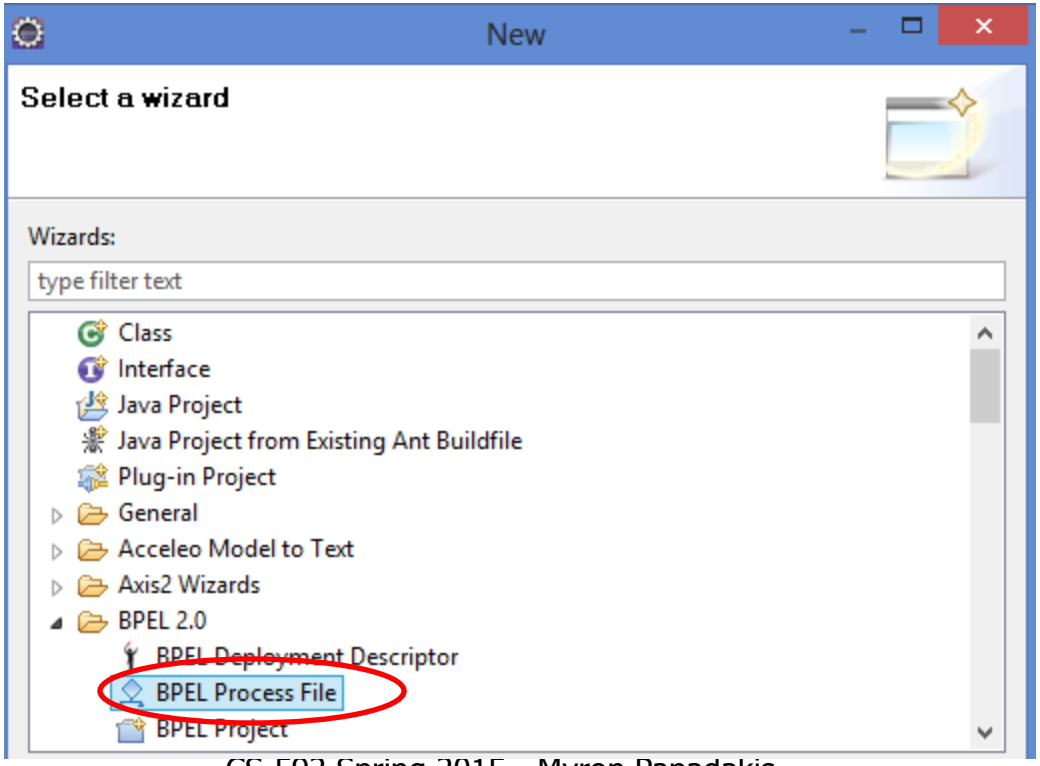
Eclipse BPEL Project



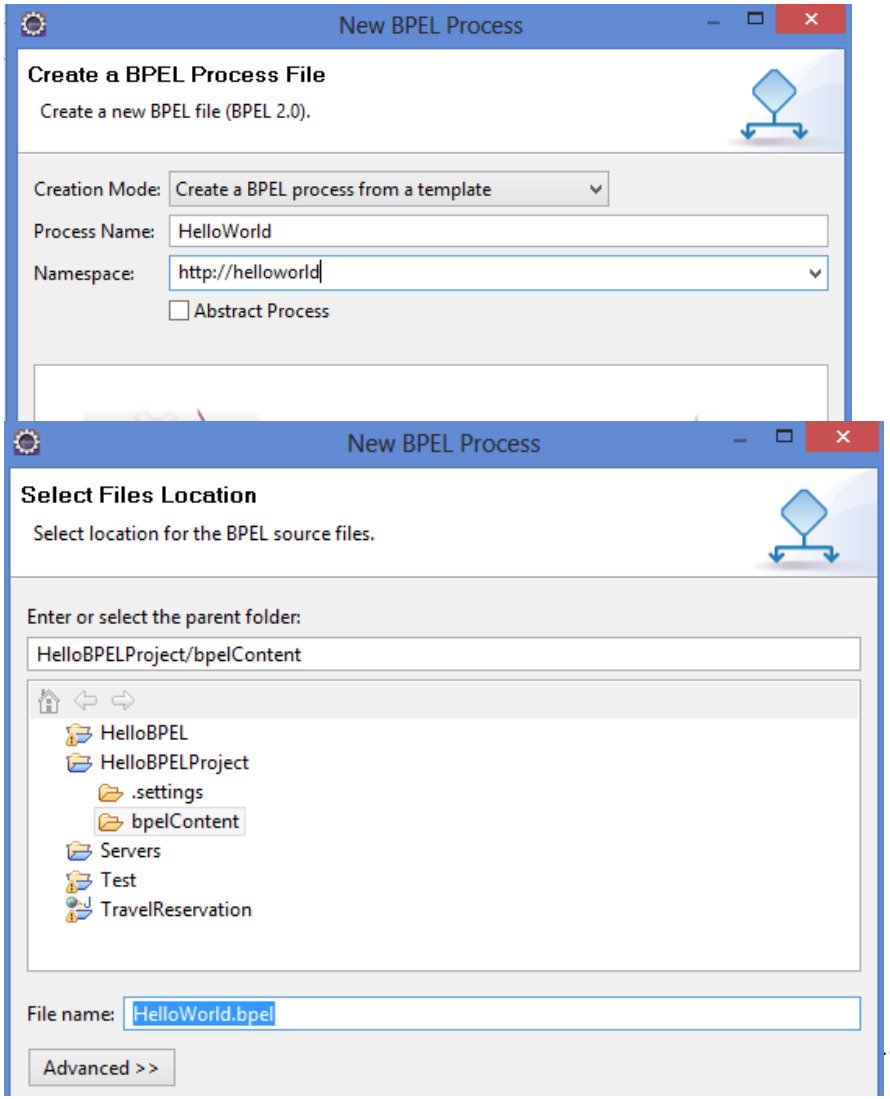


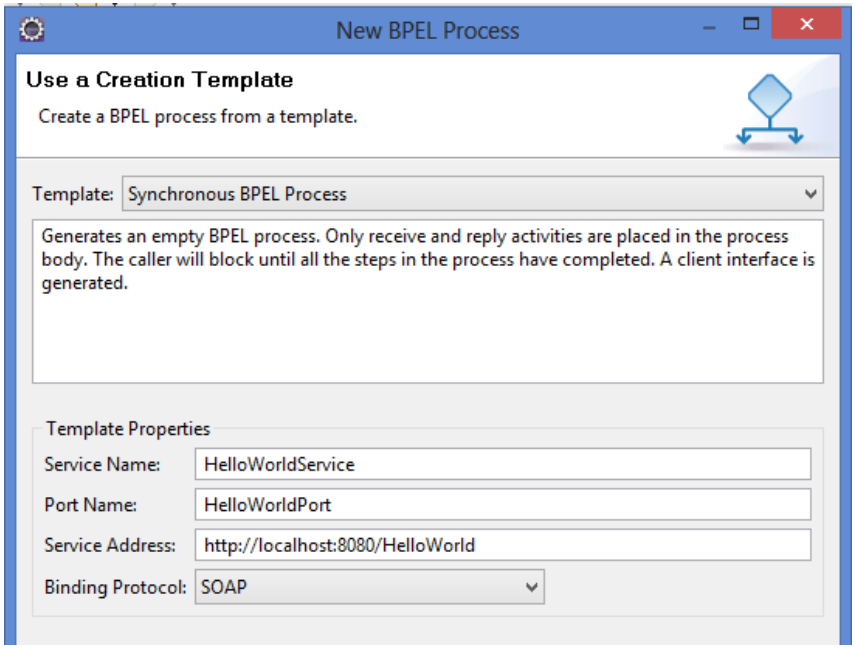
BPEL File

- Right click on the folder of your choice (i.e /bpelcontent)
  - ***New→Other→New BPEL Process File"***

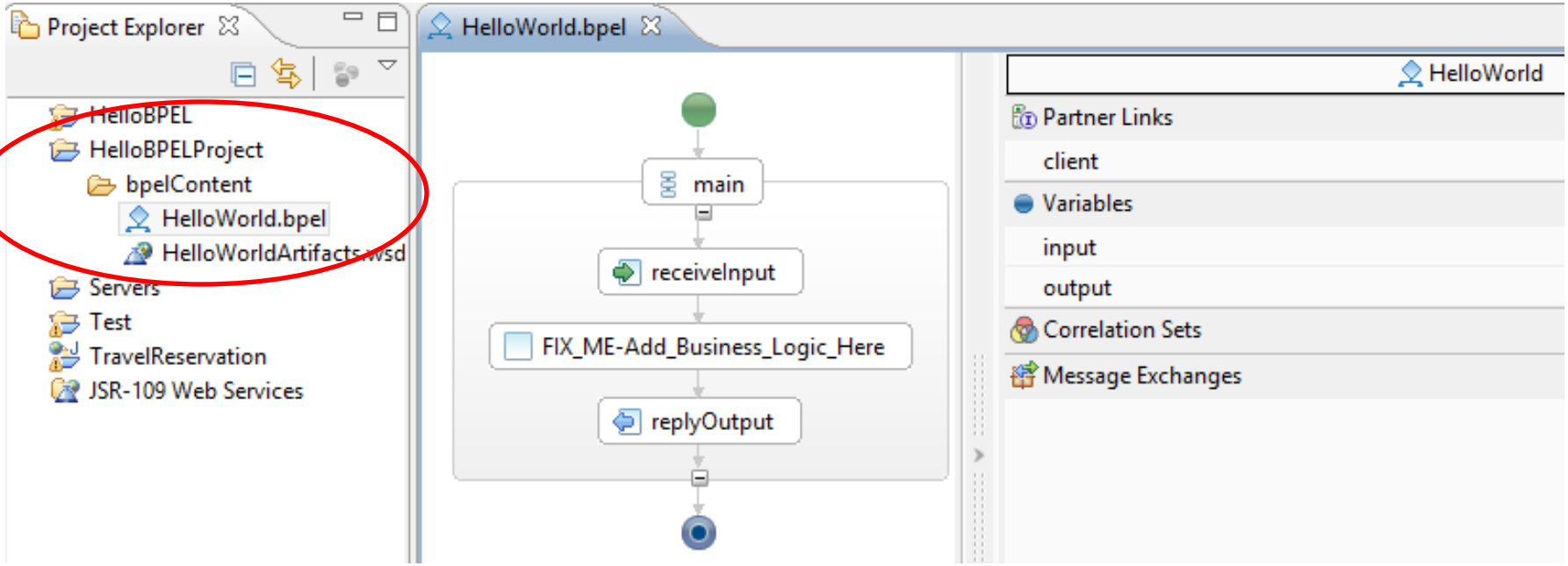


BPEL File

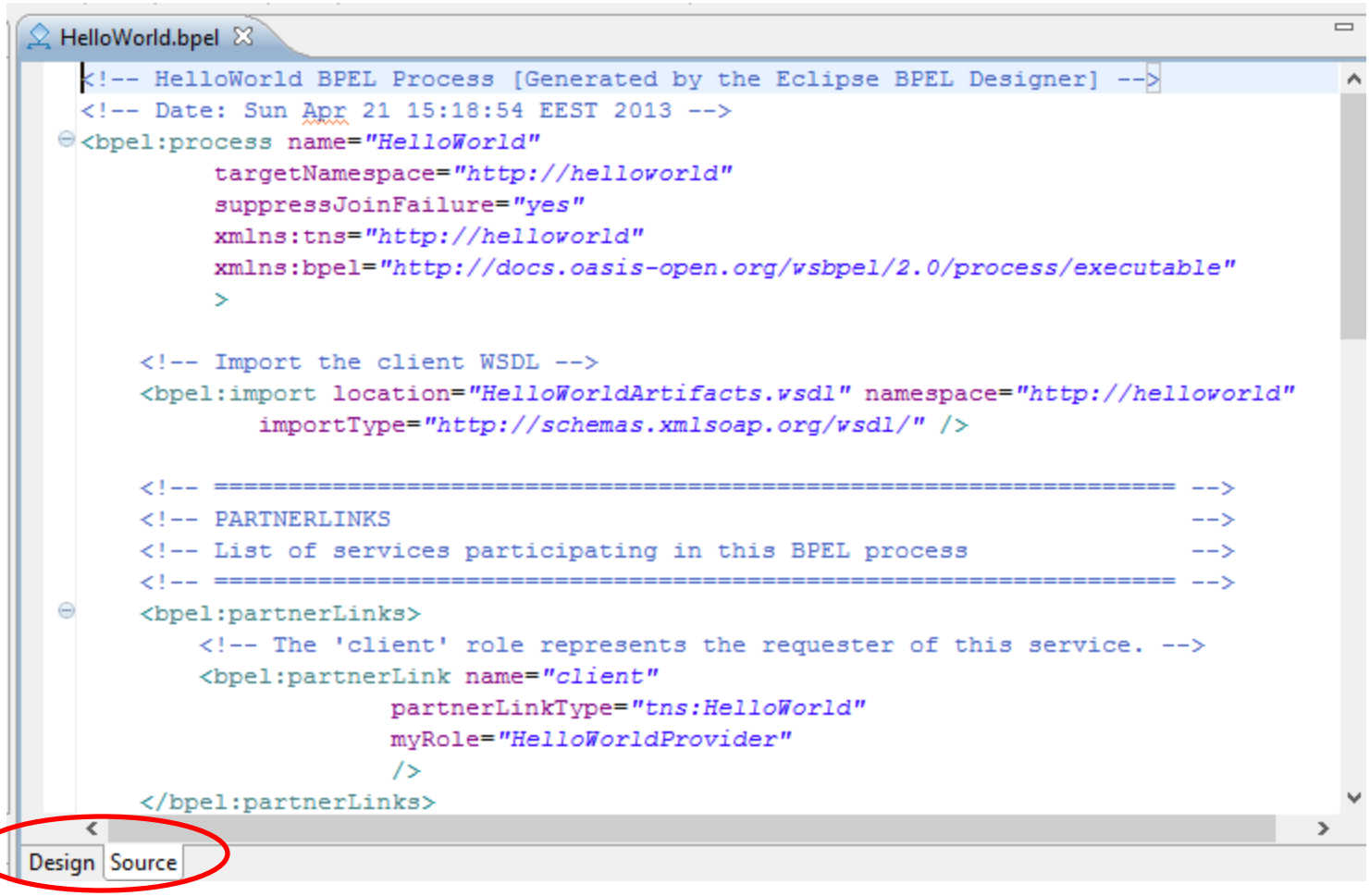

New BPEL Process
Create a BPEL Process File
Create a new BPEL file (BPEL 2.0).
Creation Mode: Create a BPEL process from a template
Process Name: HelloWorld
Namespace: http://helloworld
Abstract Process
New BPEL Process
Select Files Location
Select location for the BPEL source files.
Enter or select the parent folder:
HelloBPELProject/bpelContent
HelloBPEL
HelloBPELProject
.settings
bpelContent
Servers
Test
TravelReservation
File name: HelloWorld.bpel
Advanced >>


New BPEL Process
Use a Creation Template
Create a BPEL process from a template.
Template: Synchronous BPEL Process
Generates an empty BPEL process. Only receive and reply activities are placed in the process body. The caller will block until all the steps in the process have completed. A client interface is generated.
Template Properties
Service Name: HelloWorldService
Port Name: HelloWorldPort
Service Address: http://localhost:8080/HelloWorld
Binding Protocol: SOAP

BPEL File

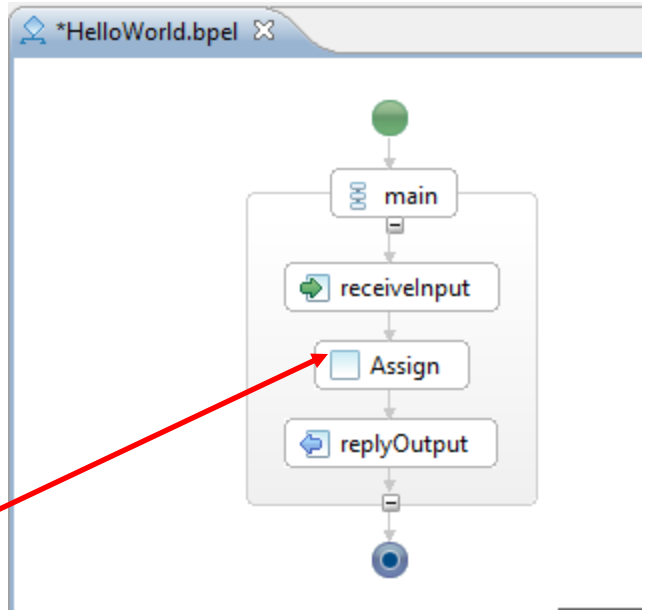

Project Explorer
HelloBPEL
HelloBPELProject
bpelContent
HelloWorld.bpel
HelloWorldArtifacts.wsd
Servers
Test
TravelReservation
JSR-109 Web Services
HelloWorld.bpel
main
receiveInput
FIX_ME-Add_Business_Logic_Here
replyOutput
HelloWorld
Partner Links
client
Variables
input
output
Correlation Sets
Message Exchanges

BPEL File


HelloWorld.bpel
<!-- HelloWorld BPEL Process [Generated by the Eclipse BPEL Designer] -->
<!-- Date: Sun Apr 21 15:18:54 EEST 2013 -->
<bpel:process name="HelloWorld"
targetNamespace="http://helloworld"
suppressJoinFailure="yes"
xmlns:tns="http://helloworld"
xmlns:bpel="http://docs.oasis-open.org/wsbpel/2.0/process/executable"
>
<!-- Import the client WSDL -->
<bpel:import location="HelloWorldArtifacts.wsdl" namespace="http://helloworld"
importType="http://schemas.xmlsoap.org/wsdl/" />
<!-- === -->
<!-- PARTNERLINKS -->
<!-- List of services participating in this BPEL process -->
<!-- === -->
<bpel:partnerLinks>
<!-- The 'client' role represents the requester of this service. -->
<bpel:partnerLink name="client"
partnerLinkType="tns:HelloWorld"
myRole="HelloWorldProvider"
/>
</bpel:partnerLinks>
Design
Source

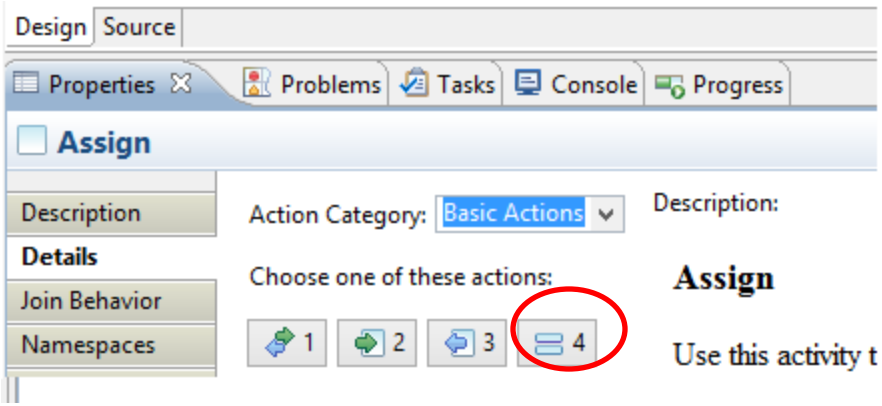
Example

- We want to design an application that responses the client the same string that it send to the service provider.
- Therefore, we have to add an activity which assign the received input to the output.
- To do this, click on the Assign activity on the Actions folder of the Palette and drag it to the space between the receiveInput and replyOutput activities (if it is not already present)
- IF it is present simply rename the assign activity from “FIX_ME-Add_Business_Logic_Here” to “Assign”

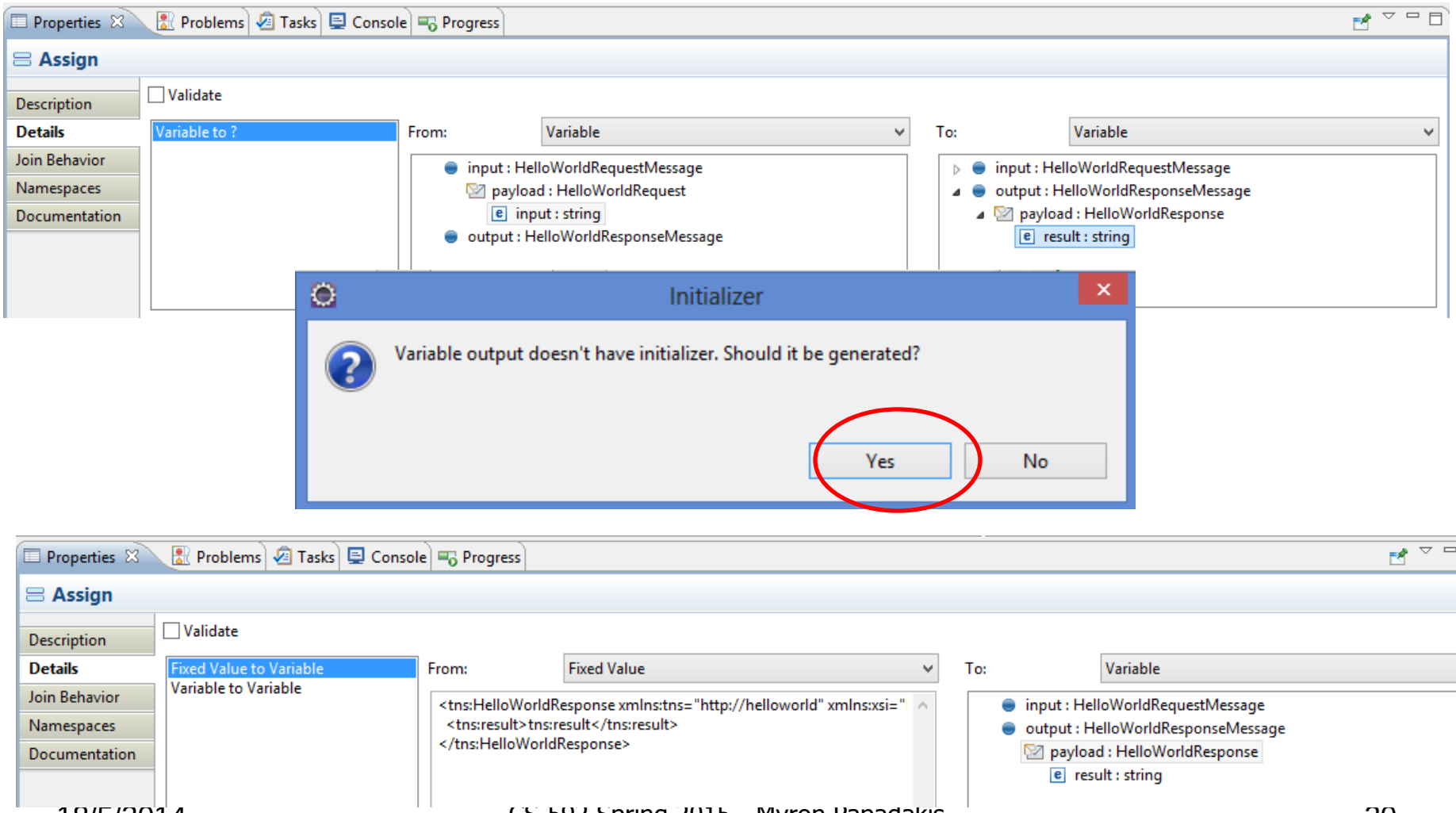


Example

- Right click on the **Assign** activity. Select **Show in properties**.
- In the Properties tab, select **Details**. Click on the **New** button to define an assign operator.
- Assign from **input->payload->input** to **output->payload->result**.
  An **Initializer** popup dialog appear. Click **Yes** to initialize the output variable and save the process.

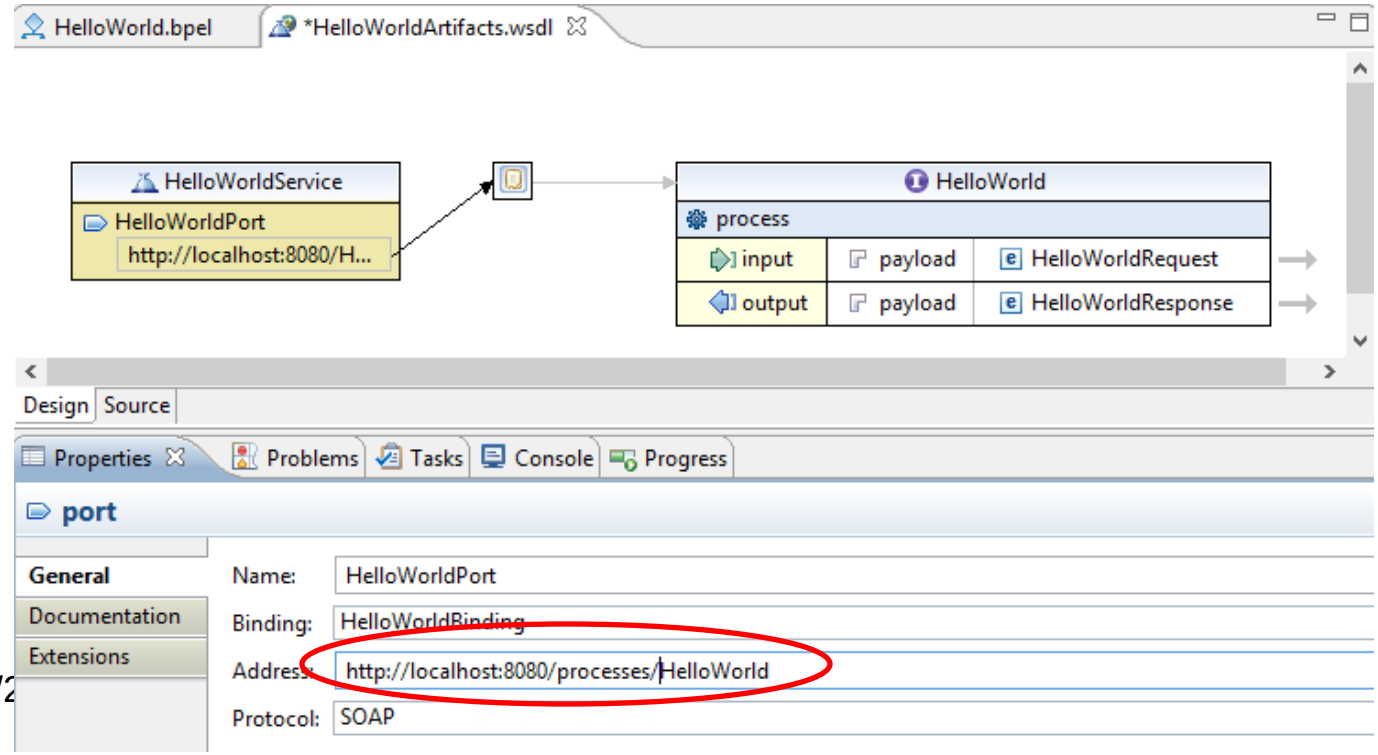


Example


Properties
Problems
Tasks
Console
Progress
Assign
Description
Details
Join Behavior
Namespaces
Documentation
Validate
Variable to ?
From:
Variable
To:
Variable
input : HelloWorldRequestMessage
payload : HelloWorldRequest
input : string
output : HelloWorldResponseMessage
input : HelloWorldRequestMessage
output : HelloWorldResponseMessage
payload : HelloWorldResponse
result : string
Initializer
Variable output doesn't have initializer. Should it be generated?
Yes
No
Properties
Problems
Tasks
Console
Progress
Assign
Description
Details
Join Behavior
Namespaces
Documentation
Validate
Fixed Value to Variable
Variable to Variable
From:
Fixed Value
To:
Variable
<tns:HelloWorldResponse xmlns:tns="http://helloworld" xmlns:xsi="
<tns:result>tns:result</tns:result>
</tns:HelloWorldResponse>
input : HelloWorldRequestMessage
output : HelloWorldResponseMessage
payload : HelloWorldResponse
result : string

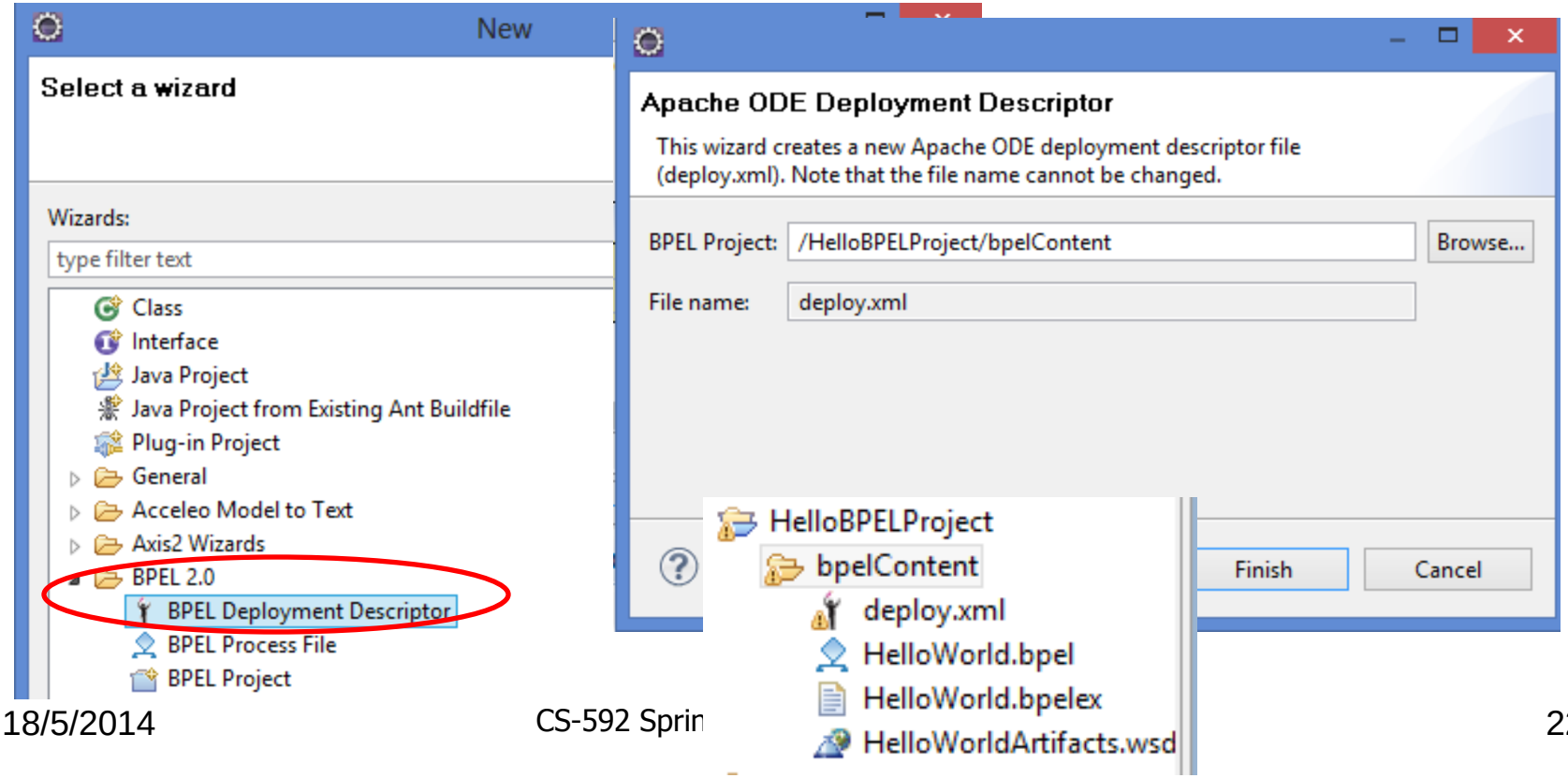
WSDL

- The BPEL process is designed.
- Now we have to define the Service, Binding type and PortType address to run the HelloWorld service.
- Right click on the **HelloWorldArtifacts.wsdl** file, select **Open With->WSDL Editor**.



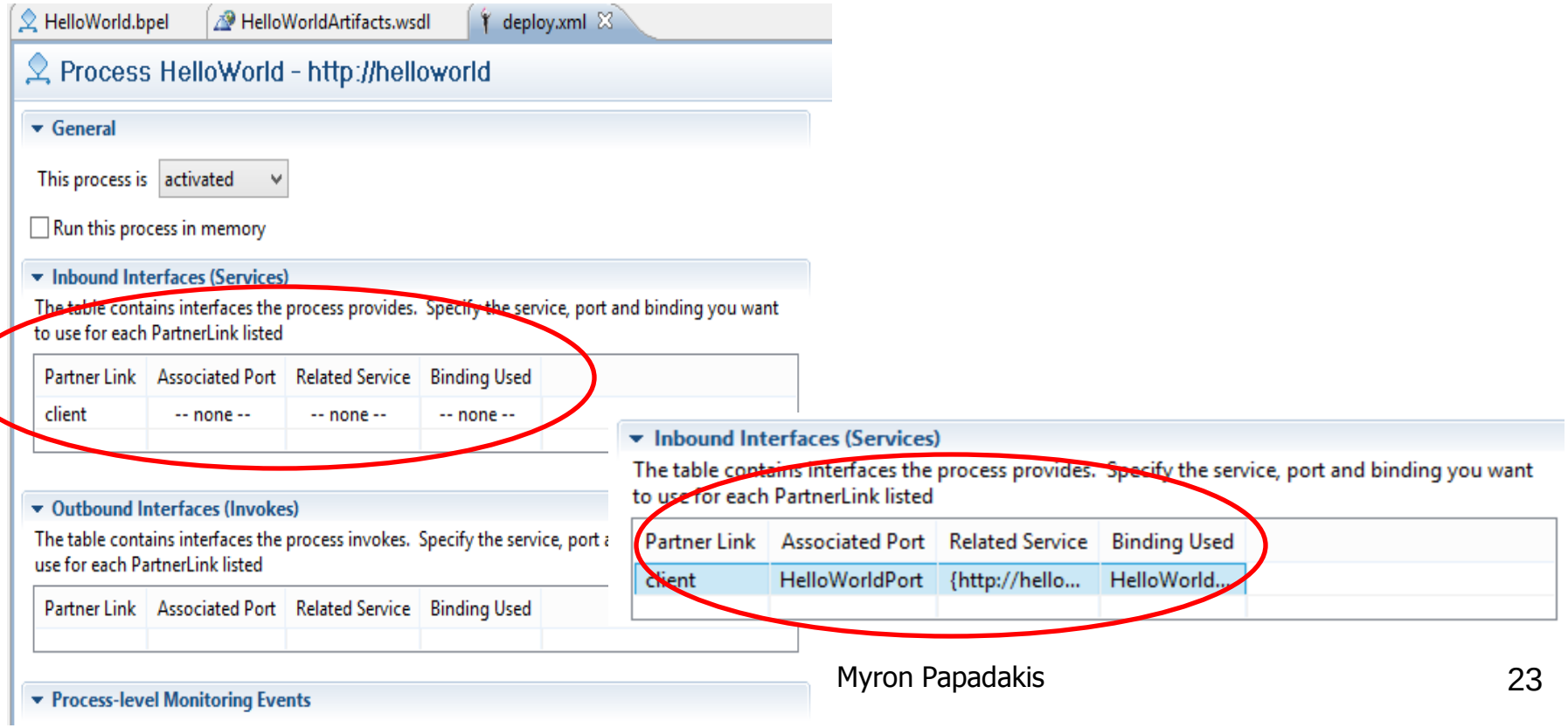
Deploy BPEL in Apache Ode

- To let the Apache ODE know your declarations and designs.
- You have to add a descriptor. Right click on the BPEL_HelloWorld project, select **New->Others->BPEL 2.0->Apache ODE Deployment Descriptor**.



Deploy BPEL in Apache Ode

- In the Inbound Interfaces table, select the Partner Link as **client**, Associate Port as **HelloPort**. Click on Related Service then other fields will be automatically filled. Save the "deploy.xml" file.



Time to run the process

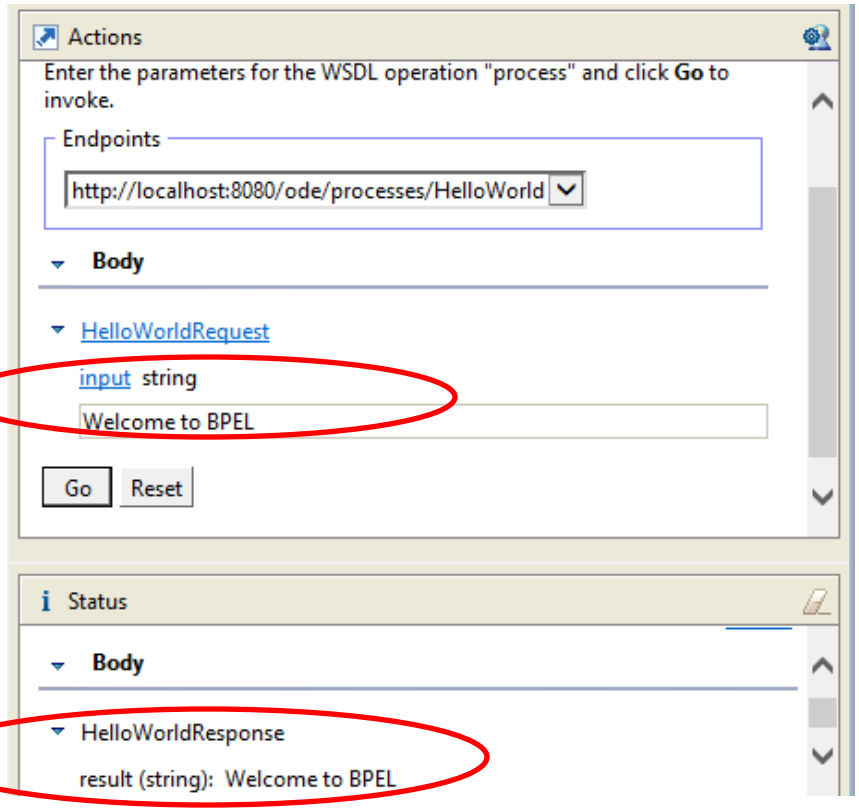
- Now you have finished your designs and it's time to run the process.
- However, the ODE Server on the Eclipse could not recognize your BPEL process because it is designed after the server's set up.
- Therefore you have to **restart your Eclipse** (File → Restart)
- After the restart, right click on the **Ode v1.x Server at localhost**, select **Add and Remove Projects**

- Select the **BPEL_HelloWorld/HelloWorld** in the Available projects box and click **Add**. Then, click **Finish**.
- Start the ODE server. If the deployment was successful, you will something similar to the following output in the console tab.

```
15:57:19,709 INFO  [DeploymentPoller] Deployment of artifact HelloBPELProject successful: [{http://helloworld}HelloWorld-6]
```

Testing the process

- We will test the "Hello World" deployment with Web Services Explorer provided by Eclipse.
- Right click on the **HelloWorldArtifacts.wsdl** file. Select **Web Services->Test with Web Services Explorer**.
- Select the operation **process**. Then, type **”Welcome to BPEL”** in the input textbox.
- If your deployment was success, you will get the same string in the SOAP response

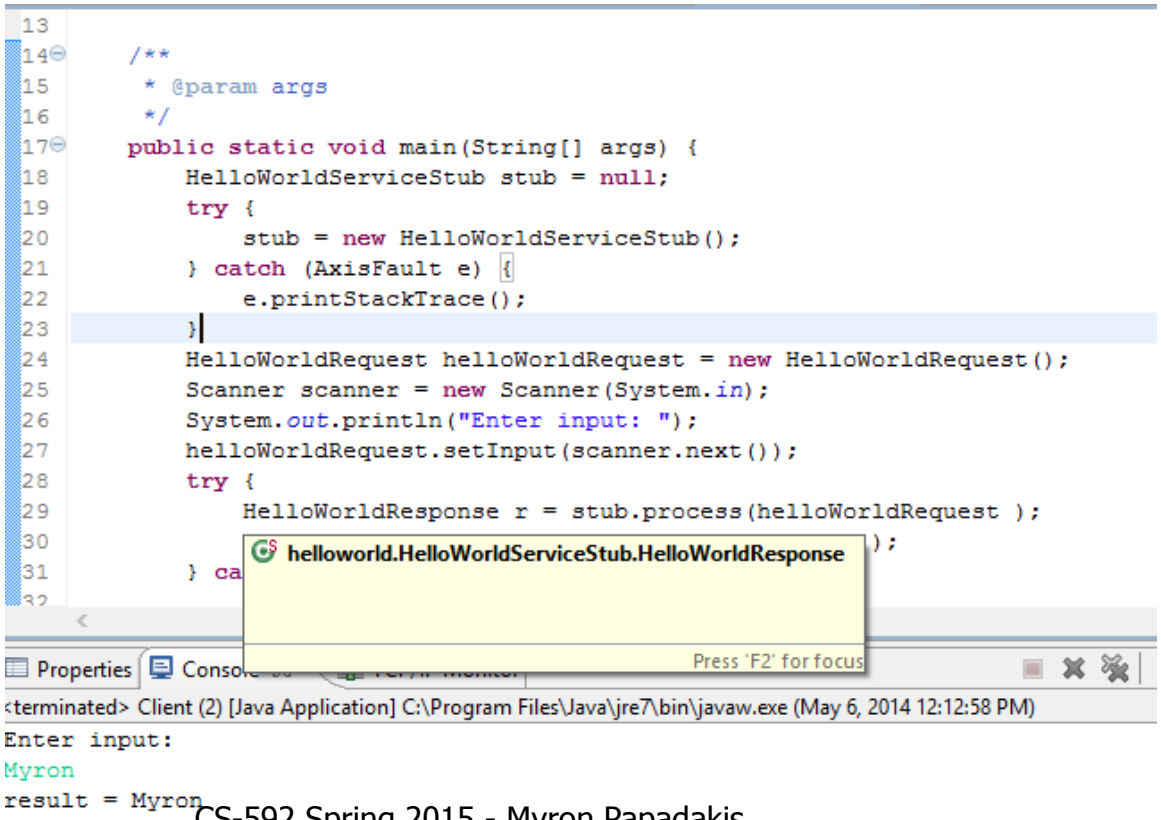


Option 2

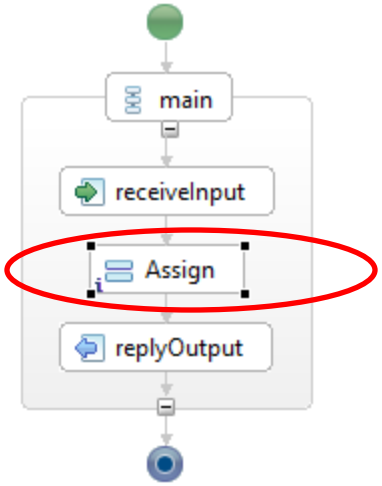
- Deploy to ODE in Tomcat
- Create a folder in **tomcat/webapps/ode/WEB-INF/processes**
- Copy the **.bpel** file, **.wsdl** file and the ODE **deploy.xml**
- Restart Tomcat and visit **http://localhost:8080/ode**
- An extra file will be created with the extension **.deployed** inside

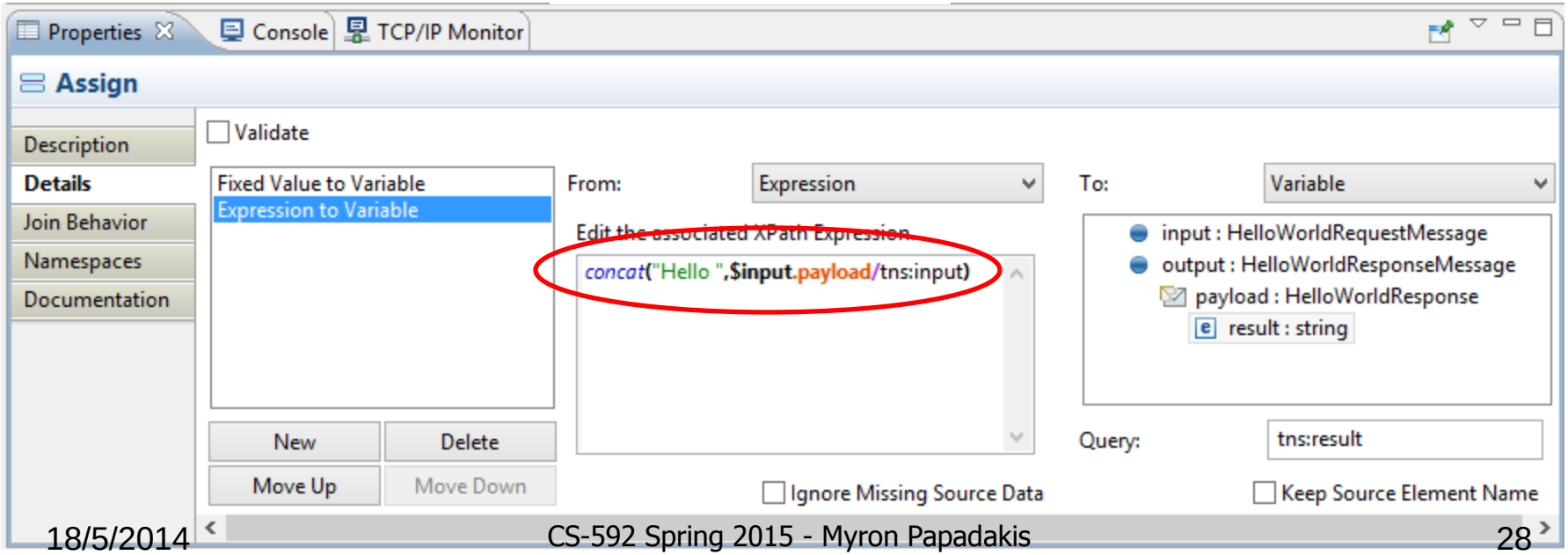
Clients

- Since the bpel process is itself a web service, clients can be created in the same way as demonstrated in the previous labs (stubs,etc)

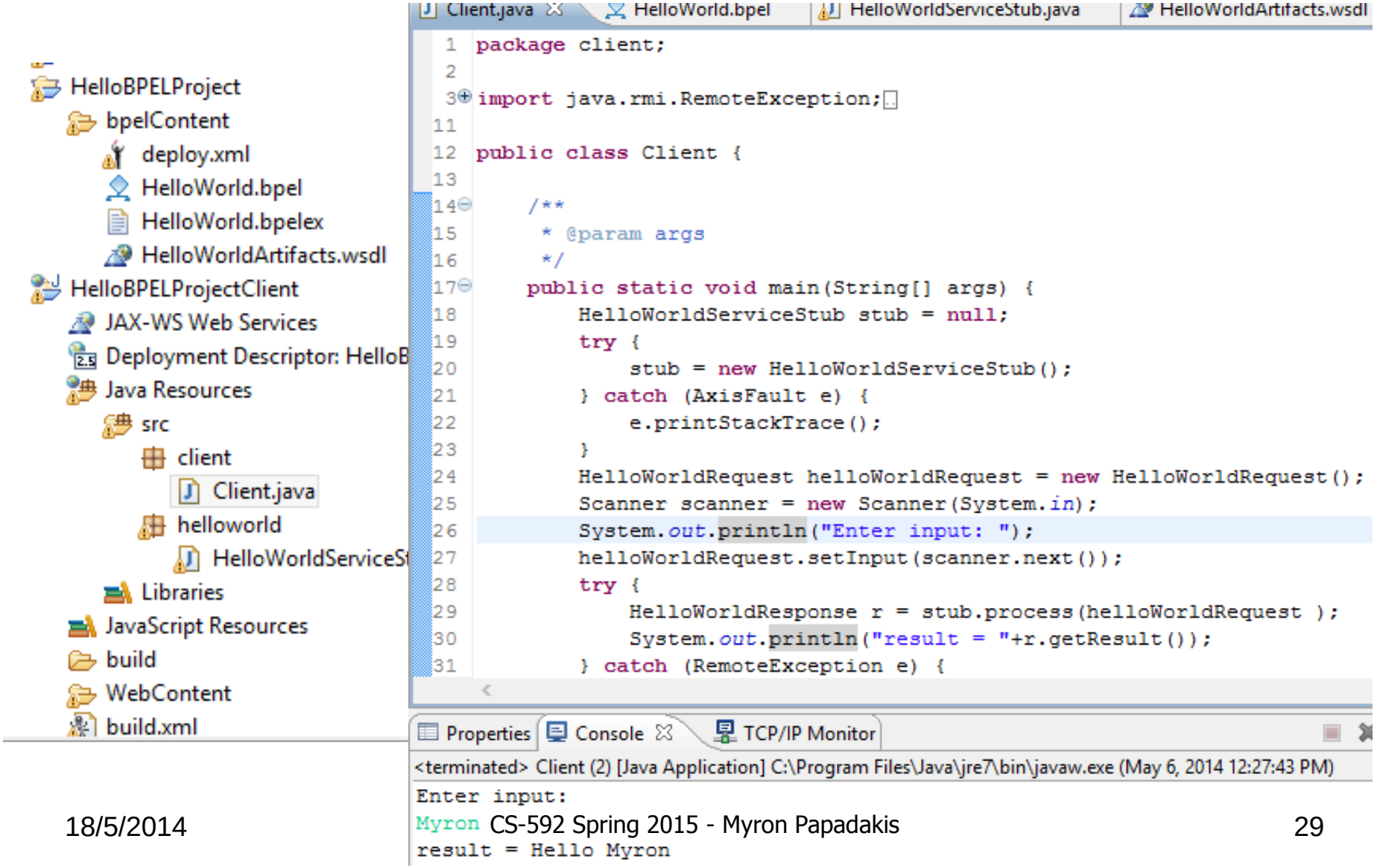


Modifying the Hello BPEL Project





Modifying the Hello BPEL Project



References

- http://www.eclipse.org/bpel/users/pdf/HelloWorld-BPELDesignerAndODE.pdf
- http://www-inf.int-evry.fr/cours/WebServices/TP_BPEL/helloworld.html
- http://www.soapui.org/Working-with-soapUI/getting-started.html

Τέλος Ενότητας

Χρηματοδότηση

•Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.

•Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Κρήτης**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.

•Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.

Σημειώματα

Σημείωμα αδειοδότησης

- Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».

[1] http://creativecommons.org/licenses/by-nc-nd/4.0/

•Ως **Μη Εμπορική** ορίζεται η χρήση:

–που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο

–που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο

–που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

•Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

.

Σημείωμα Αναφοράς