ORACLE

Oracle Workshop for WebLogic 10g R3 Hands on Labs

Workshop for WebLogic extends Eclipse and Web Tools Platform for development of Web Services, Java, JavaEE, Object Relational Mapping, Spring, Beehive, and Web Applications. It installs as a plug-in to your existing Eclipse, or will install Eclipse for you.

Workshop for WebLogic is used to develop, build, assemble, deploy, debug and test Java SE, Java EE, Web Services, Java Web Applications, Object Relational Mapping on Eclipse. If you are an Oracle WebLogic Server user, this is the free tool for you.

10g R3 Release Notes

Workshop for WebLogic introduces new tools in support of Java EE 5.0 standards. The support for Java EE5 includes the following technologies:

Java EE5 Standards Support

- Servlet 2.5
- JSP 2.1
- JSF 1.2
- JSTL 1.2
- Unified Expression Language
- JAX-WS
- JAXB 2.0
- EJB 3
    - EE5 EAR
    - EJB 3 Session Beans
    - EJB 3 Message Driven Beans

Built on Eclipse 3.3.2 and Web Tools Platform 2.0.3

Workshop for WebLogic version 10g R3 is built on the Eclipse Platform, an open source framework that is now widely used for Java development. Workshop for WebLogic extends **Eclipse 3.3.2** and the Web Tools Platform **2.0.3.**

Supported by Windows Vista

Workshop for WebLogic version 10g R3 is supported by **Windows Vista**.

XML Beans

Workshop for WebLogic version 10g R3 supports **XMLBeans 2.3**.

Workshop for WebLogic IDE Launcher

The **WorkSpace Studio launcher** has been discontinued. The Workshop for WebLogic IDE launcher is WORKSHOP_HOME/workshop[.exe].

Getting Familiar with Workshop for WebLogic 10g R3

The core components of this Eclipse based development environment are defined by the following functional areas:

- Enhanced server plug-ins for multiple versions of Oracle WebLogic Server
- Visual Oracle WebLogic Server Web Service and XML IDE
- WYSIWYG Web and presentation tier tools for portable Java Web applications
- Object-relational mapping workbench and database tools
- Apache Beehive IDE for Java Page Flow and controls
- AppXRay support for the above components
- Spring IDE Project and Spring code generation wizards
- Core IDE features for Java SE and Java EE
- Built in Web Application and Web Service test client
- Upgrade tools for Workshop 8.1, 9.2 and 10 users

Getting hands on with Workshop for WebLogic 10g R3

LAB1: Migrate a 9.2 application to 10g R3

Workshop for WebLogic provides source code migration tools for all source artifacts (source files, application, and project). Some upgrade cases cannot be automated and require developer intervention. The core upgrade harness in all cases provides some base capabilities:

- Pre-upgrade reporting, restart failed/partial upgrades, doesn't affect original source
- Will handle project structure upgrade, file extension renaming
- Will invoke domain upgrader if needed for 8.1.4 domain
- Command line version available

This lab will also introduce you to the WebLogic Runtime Migration Wizard, which can move a project (and related projects) from targeting a WLS 9.2 runtime to WLS 10g R3 runtime in one shot, forward migration only.
This automatically accomplishes several steps are that required to move the appropriate facet versions, the target runtime, and all related projects at once (e.g. web apps in an EAR).
While this can be done manually, it's tricky work to:

- disassociate the projects from the target runtime
- move facet versions (knowing which ones got to which version)
- repeat for all projects in an EAR (or referenced)
- Re-set the target runtime for each project to 10g R3

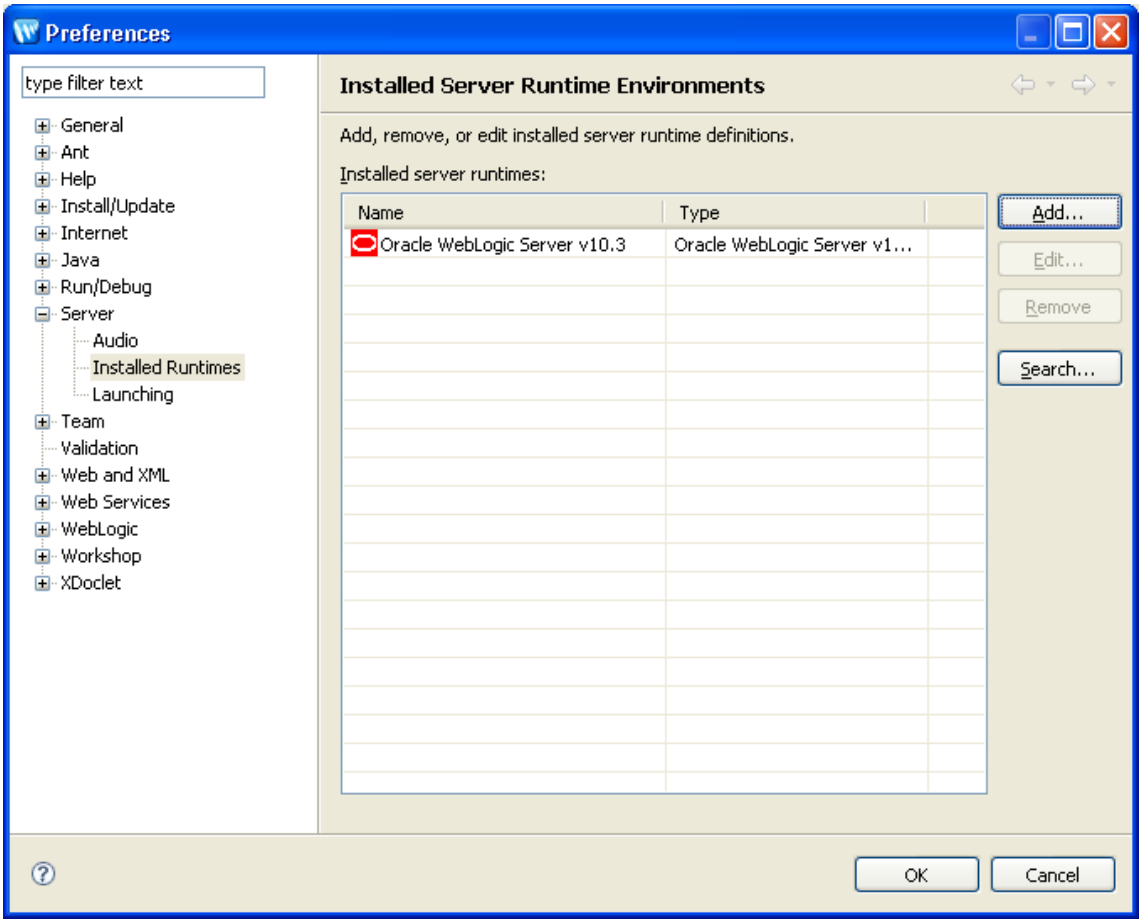
Prerequisite:

- WLS 92MP3 installed. Download it now.
- WLS 103MP0 installed. Download it now.
- The Workshop for WebLogic 9.2 projects must be saved with only the standard Workshop perspective.
- TOI Boulder.zip (92MP3 App)

Steps

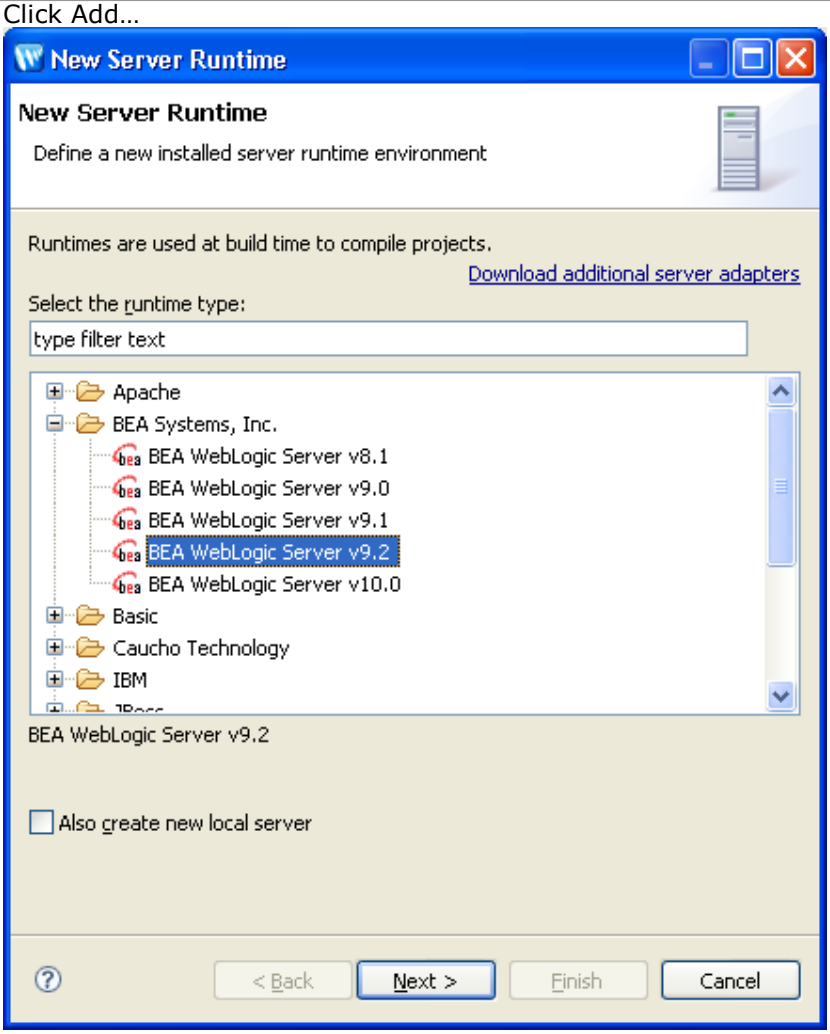
- Open Workshop for WebLogic
- Create a new workspace
- Go to Menu Window -> Preferences
-

- Select Server -> Installed Runtimes:



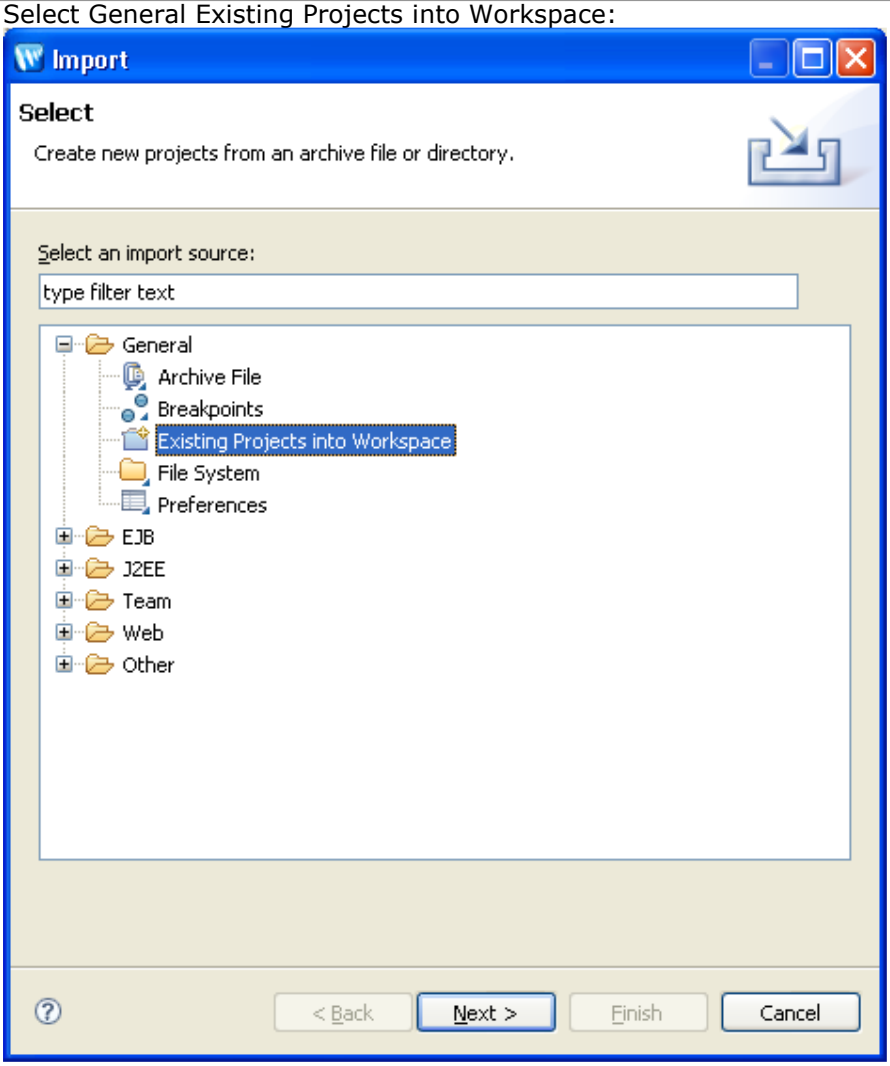
-

- Click Add…



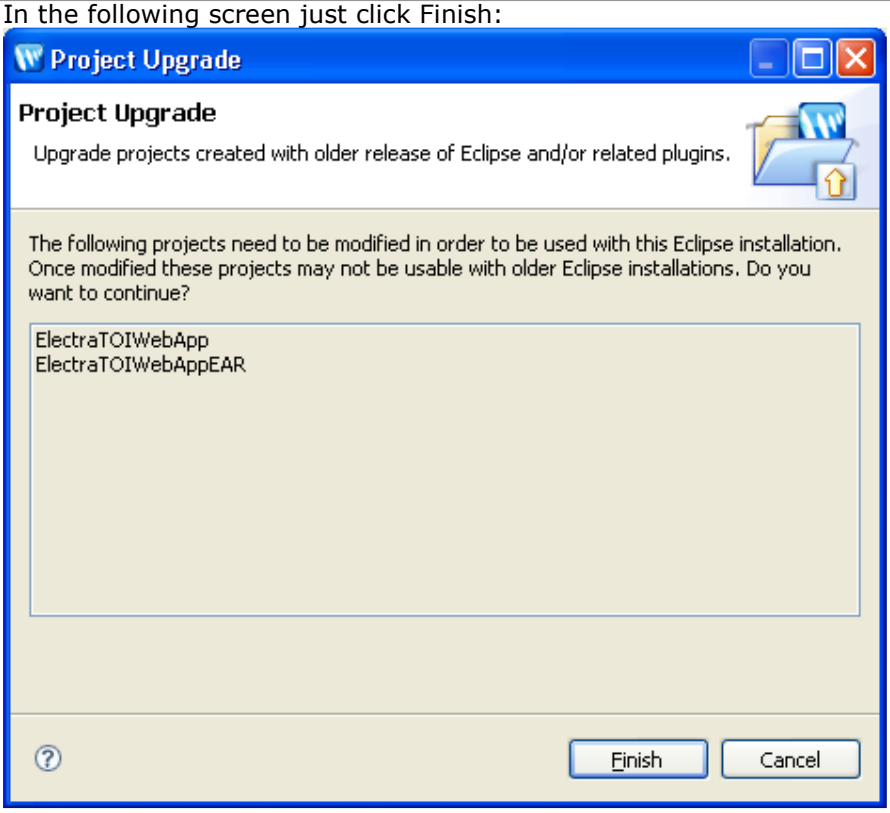
- Select BEA Systems, Inc. -> BEA Weblogic Server v9.2
- Click Next and Browse to your Weblogic 9.2 home folder
- Click Finish and OK
- Go to menu File -> Import

- Select General Existing Projects into Workspace:



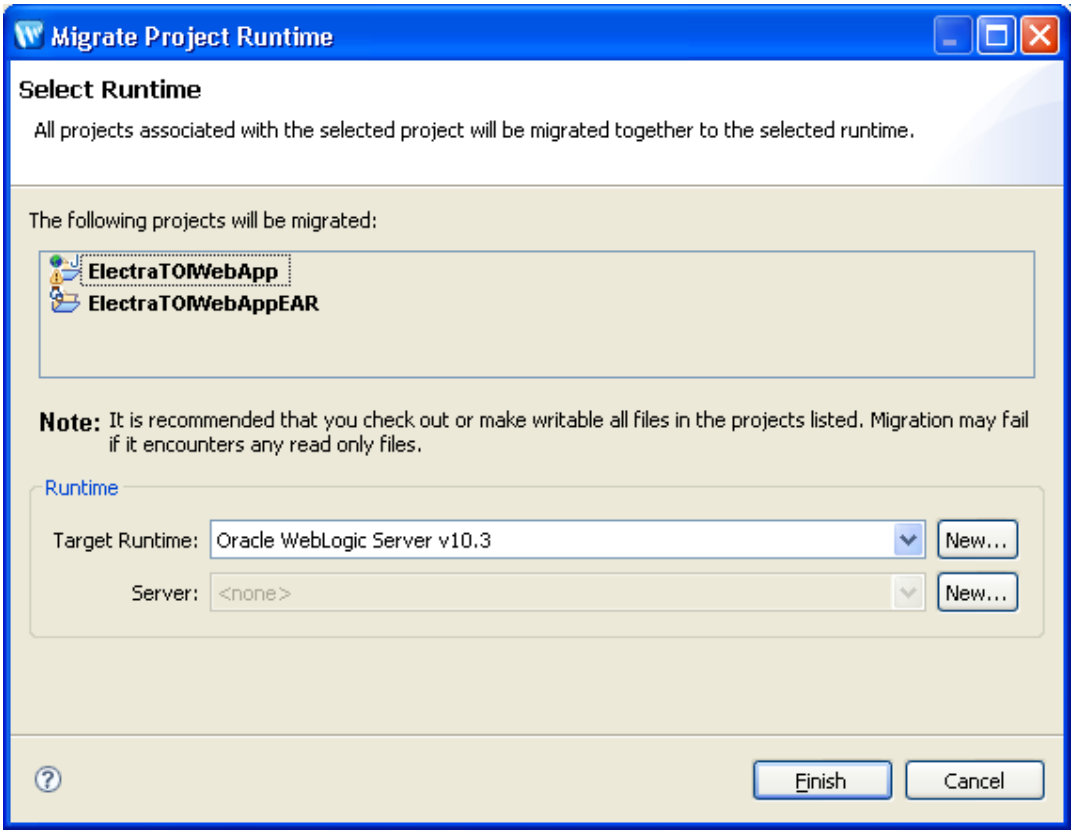
- Click Next and Select: Select Archive File
- Browse to TOI Boulder.zip file
- Click Finish

- In the following screen just click Finish:



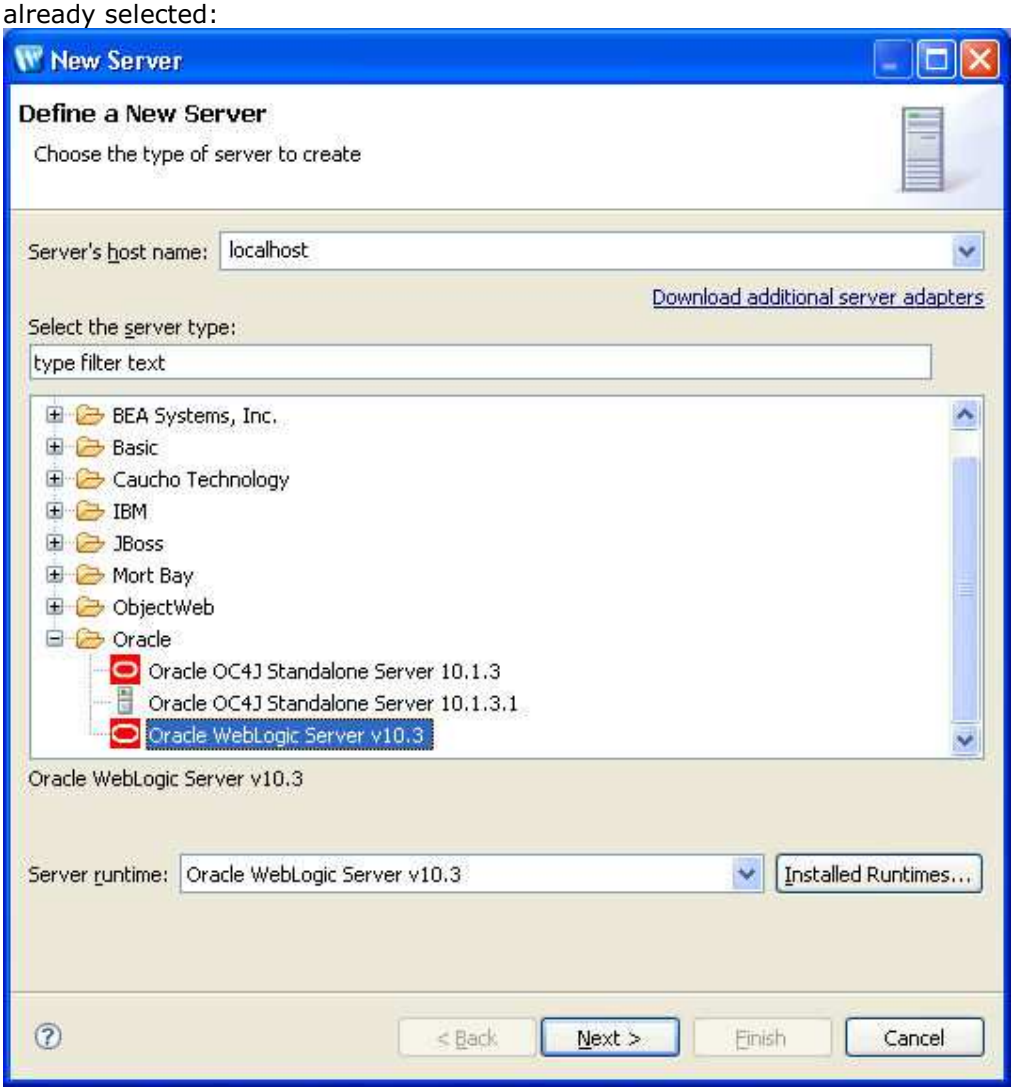
- Just wait for Workshop to finish its work of updating project and domain metadata.

- In the project view right click on the ElectraTOIWebApp and select Migrate Project Runtime:



- Click Finish
- Just wait for Workshop to rebuild its workspace.
- In Server View right click and Select New

- If all went well, in the New Server Dialog, Oracle Weblogic Server 10.3 should be already selected:



- Click Next, Create a new Workshop Domain (accept all defaults) and Browse to the Domain Home to specify it for the "New Server" Dialog.
- Click Next and click Add All
- Click Finish
- To test, go into the project view, ElectraTOIWebApp -> Java Resources -> src -> Controller.java, Right click and select Run As… -> Run On Server.

LAB2: Migrate a 8.1 application to 10g R3

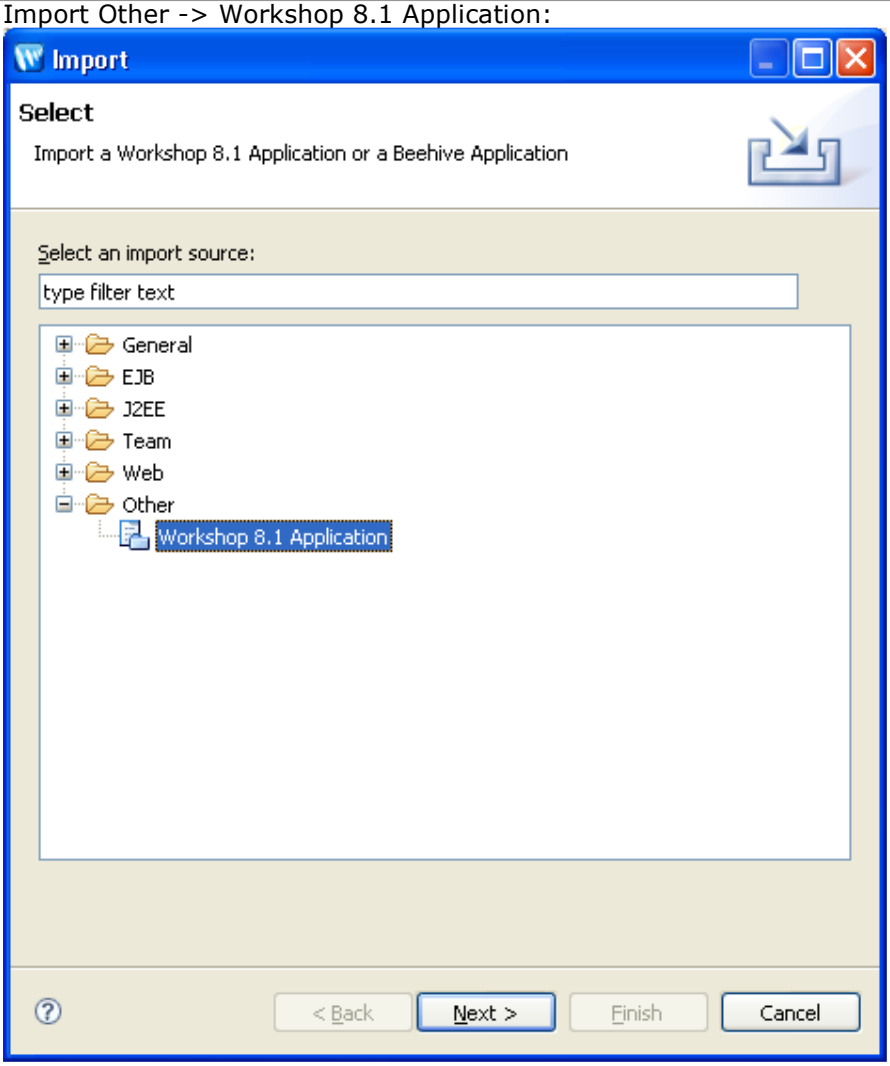
Prerequisite:

- 816App.zip

Steps

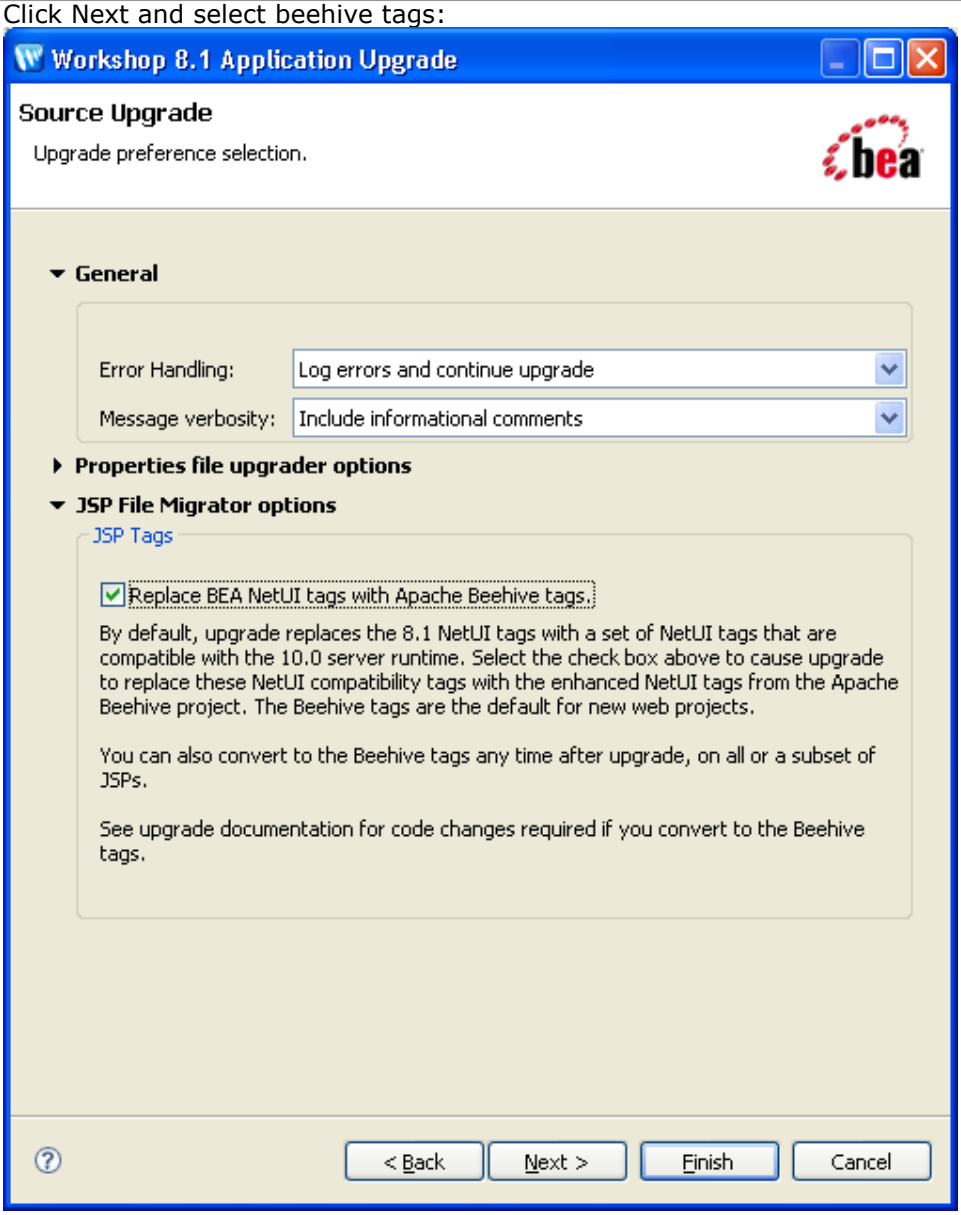
- Open Workshop for WebLogic 10g R3

- Import Other -> Workshop 8.1 Application:



- Click Next
- Browse to the 8.1 project folder and select .work file
- Chose WebLogic 10.3 for target runtime

- Click Next and select beehive tags:



- Click Next / Finish
- Replace in Appplication81Web -> WebContent -> ShowResult.jsp <netui:content value="${pageScope.Result.value}"/> by <netui:content value="${requestScope.Result.value}"/>
- Everything is ready to be deployed on 10g R3

LAB3: Secure access to the Web App

Objective: **Use WebLogic deployment descriptor editors to secure app**

Prerequisite:

- Migration Lab finished

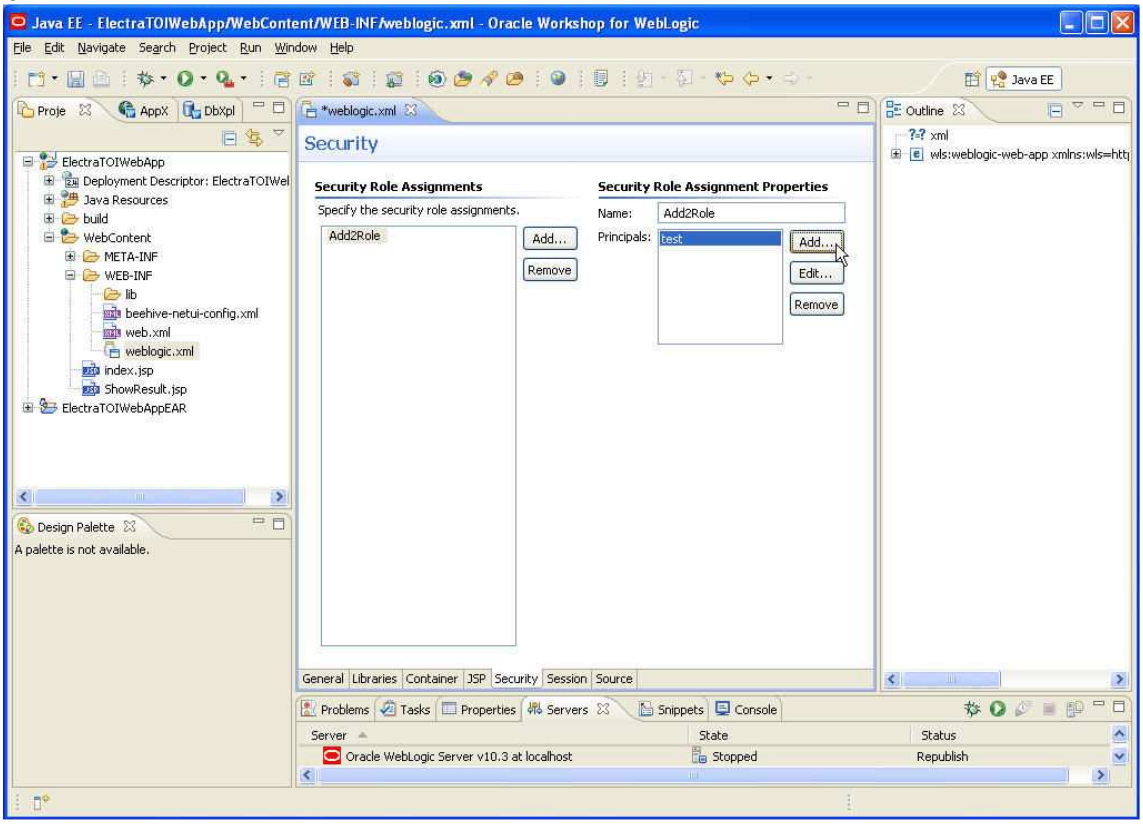
Steps

In the previously created Workshop Domain:

- Start the server
- Open WLS Console: http://localhost:7001/console
- Security Realms -> myrealm -> Users and Groups -> Users -> New
- Name: test
- Password: weblogic (you will need this again in lab 5, take a note of it)
- OK

In the previously migrated project:

- Open the weblogic deployment descriptor editor by double clicking Project view -> ElectraTOIWebApp -> WebContent -> WEB-INF -> weblogic.xml
- Security tab -> Add…
- Name: Add2Role
- Principals: Add …
- Principal: test ->
  OK



- Then double click to open Java standard deployment descriptor, web.xml
- Add the following in the source view before the closing web-app tag (The GUI does not yet permit to edit those fields using wizard like with weblogic.xml)

```
<security-constraint>
        <web-resource-collection>
```

```
                    <web-resource-name>Secure</web-resource-name>
                    <url-pattern>*</url-pattern>
                    <http-method>GET</http-method>
                    <http-method>POST</http-method>
                </web-resource-collection>
                <auth-constraint><description></description>
                <role-name>Add2Role</role-name>
                </auth-constraint>
    </security-constraint>
    <login-config><auth-method>BASIC</auth-method></login-config>
    <security-role><role-name>Add2Role</role-name></security-role>
```

- Now only user test should be able to access the WebApp. Deploy and run the app to check the access control.

LAB4: Remote server deployment

Objective: Target and deploy our migrated application to a development cluster using Workshop for WebLogic. This uses a remote connection for what is actually local cluster/machine.

Prerequisite:

- Migration Lab finished

Steps

Launch Workshop for WebLogic with the workspace that you used for Lab 1, which upgraded the "ElectraTOIWebApp".

Create a new Workshop 2 nodes cluster production domain using 10g R3 config wizard.

- In config wizard:
- Select "Create a new WebLogic domain", click Next
- Select "Generate a domain ..." and check Workshop for WebLogic 10g R3 component, click Next
- For User Name and Password use weblogic, click Next
- In Weblogic Domain Startup Mode select "Production Mode", click Next
- In "Do you want to customize..." select Yes, click Next
- Click Next again
- In Admin Server configuration change listen port to 8001, click Next
- In Configure Managed Servers, add 2 managed severs n1 with port 8101 and n2 with port 8201, click Next
- In Configure Clusters, add one cluster named c1, click Next
- In Assign Servers to Clusters, add n1, n2, click Next
- In Configure Machines, add one machine named m1, click Next
- In assign Servers to Machines, add AdminServer, n1, n2, click Next
- Click Next in all the remaining dialogs until you are asked for domain name and location, choose the name and location you want. Accepting the default location and changing the default domain will keep everything in the same directory, and help you differentiate from other domains.

Start the Admin and Managed Servers.

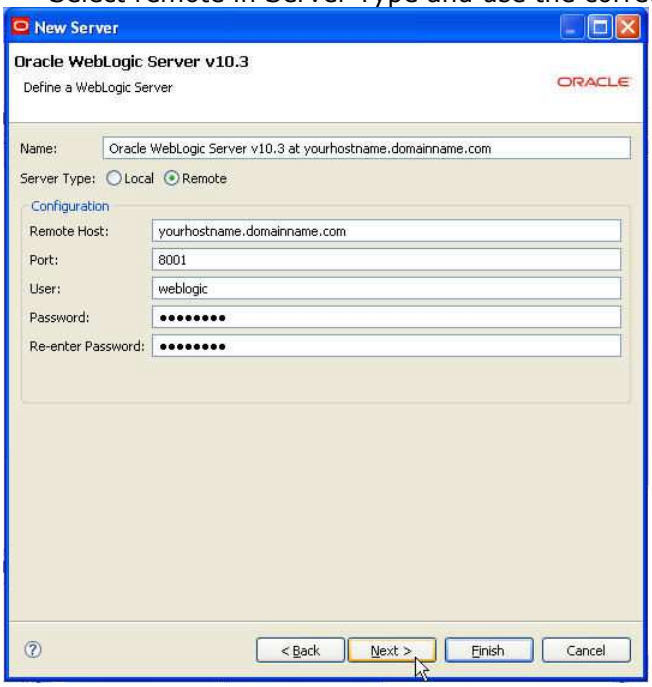
- Open 3 command line windows so it's easy to track the server log output separately.
- Go to command line window 1 and navigate to the domain you just created. If you accepted defaults, it should be
    - <bea_home>\user_projects\domains\<yourname>_domain
- Run the admin server start script, making sure it starts without errors.
    - startWebLogic.cmd

- In the command line windows 2 and 3, change directories to <bea_home>\user_projects\domains\<yourname>_domain\bin
- Execute this command in window 2
  - startManagedWebLogic.cmd n1 http://localhost:8001
- Execute this command in window 3
  - startManagedWebLogic.cmd n2 http://localhost:8001

End state is 2 managed, one admin server running in separate DOS windows.

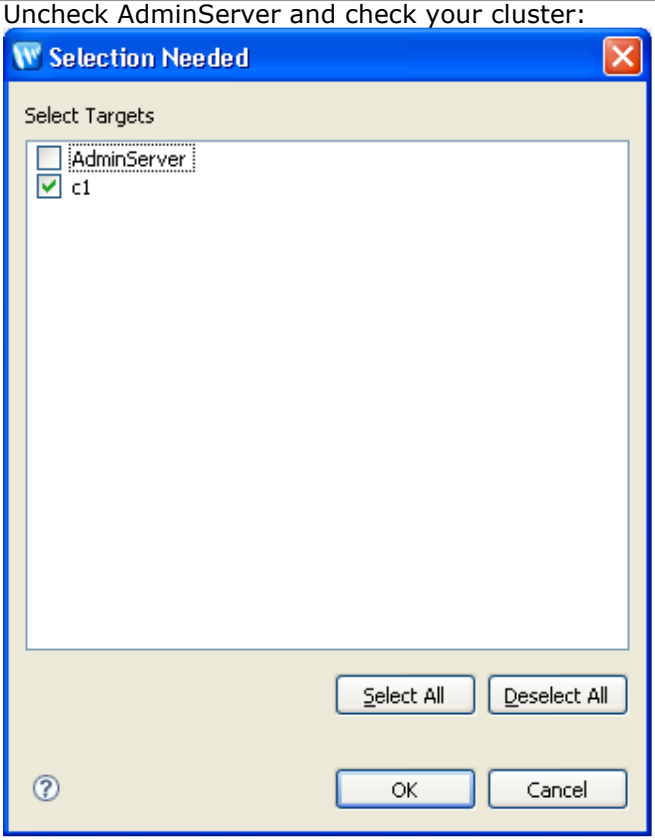
Add a remote server to workshop project.

- In Server View right click New -> Server
- Select 10.3 runtime and click Next
- Select remote in Server Type and use the correct settings in the other fields:



- Click Next
- Add All then Finish
- In the server view, double click on the newly created server to open the property sheet
- In Startup & Deployment section, click the Select button

- Uncheck AdminServer and check your cluster:



- Click OK
- Open a browser and navigate to http://localhost:7001/console. The login should be weblogic/weblogic.
- Add the test user as in the previous lab 3. (while the ElectraTOIWebApp is configured for form authentication, the new domain only has default users).
- Then test http://<host-name>:8101/ElectraTOIWebApp and http://<host-name>:8201/ElectraTOIWebApp