# Philippines AWS – Development Environment Setup Guide

Created Date: Feb 18, 2014

Created By: Kwan Swee Yee

Update History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Prepared By** | **Remarks** |
| 1.0 | Feb 18, 2014 | Kwan Swee Yee | Initial Draft |
| 1.1 | Mar 13, 2014 | Kwan Swee Yee | Revised Development Environment Setup Guide |
| 2.0 | Mar 8,2020 | Alex Miao | Update some steps use yellow color to mark |

[Philippines AWS – Development Environment Setup Guide 1](#_Toc382502871)

[1. Introduction 2](#_Toc382502872)

[2. Prerequisite 2](#_Toc382502873)

[3. Environment Setup 3](#_Toc382502874)

[Workspace Configuration 3](#_Toc382502875)

[Create AWS Enterprise Application Project 5](#_Toc382502876)

[Create EJB Module 9](#_Toc382502877)

[Create WEB Module 12](#_Toc382502878)

[Setup AWS Enterprise Application Project 16](#_Toc382502879)

[Setup EJB Module 18](#_Toc382502880)

[Setup WEB Module 26](#_Toc382502881)

[Create and Setup for File Server Enterprise Application Project with WEB module 27](#_Toc382502882)

[Create Weblogic Server 31](#_Toc382502883)

[Appendix A - Setup Weblogic and OEPE 35](#_Toc382502884)

[Installing OEPE 35](#_Toc382502885)

[Appendix B – Domain Creation 41](#_Toc382502886)

[Appendix C – Domain Setup 45](#_Toc382502887)

[AWS Domain 45](#_Toc382502888)

[File Server Domain 46](#_Toc382502889)

## Introduction

This document is to provide information about steps on how to setup PH AWS development environment under Weblogic.

## Prerequisite

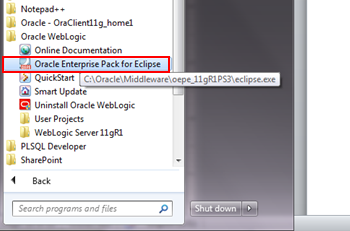
1. OEPE Eclipse and Weblogic 10.3.6 already installed in you environment. If you have not done so, kindly refer to [Appendix A](#_Appendix_A_-).
2. Source code has been checkout from svn branch. Refer to your supervisor for the svn branch.
3. Domains have been created for AWS and File Server. Kindly refer to [Appendix B](#_Appendix_B_–) and [Appendix C](#_Appendix_C_–) for domains creation and domain setup.
4. Empty Workspace folder has been created.

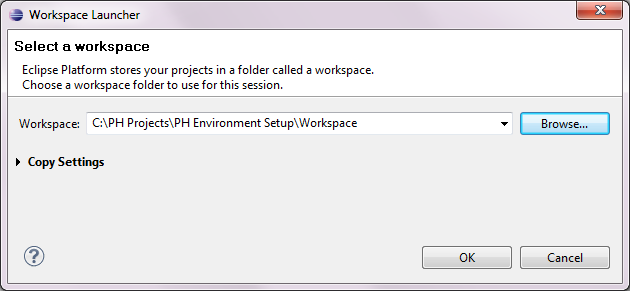
Refer below for the file structure you should have before start on environment setup.

## Environment Setup

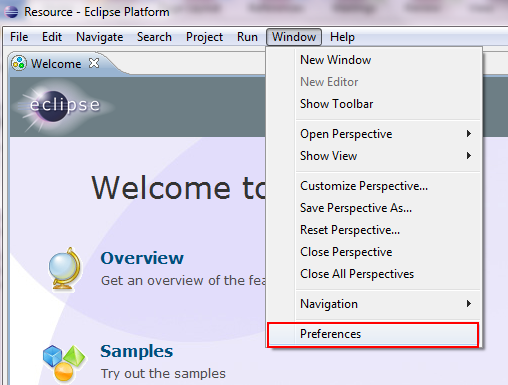
### Workspace Configuration

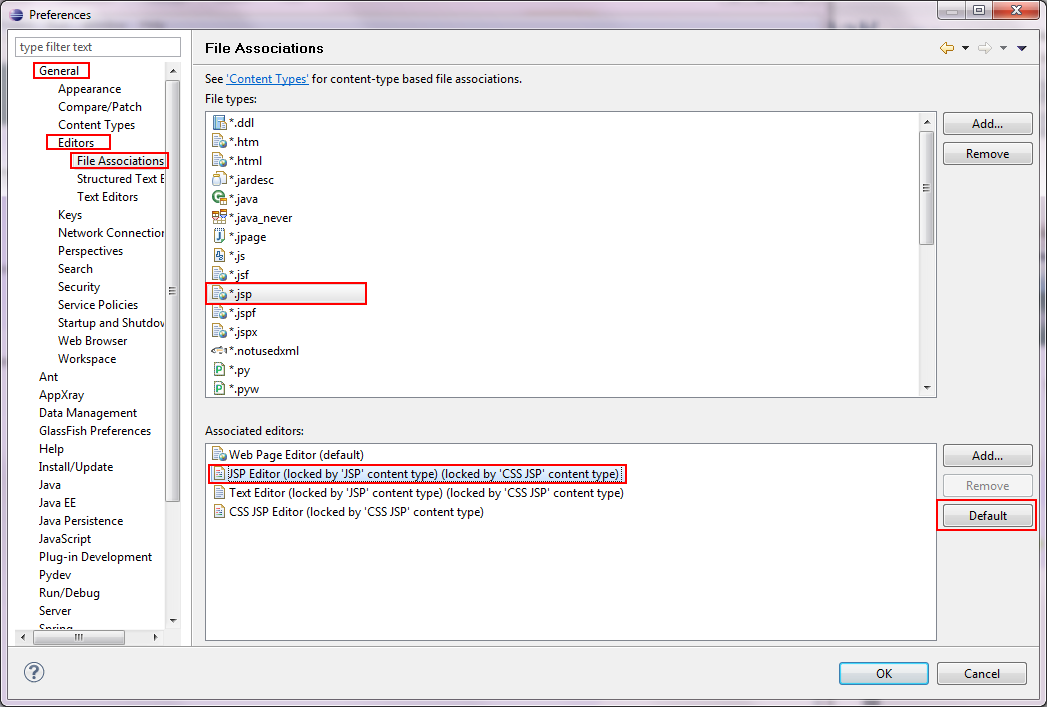
1. Open OEPE Eclipse and switch to the workspace under your project.





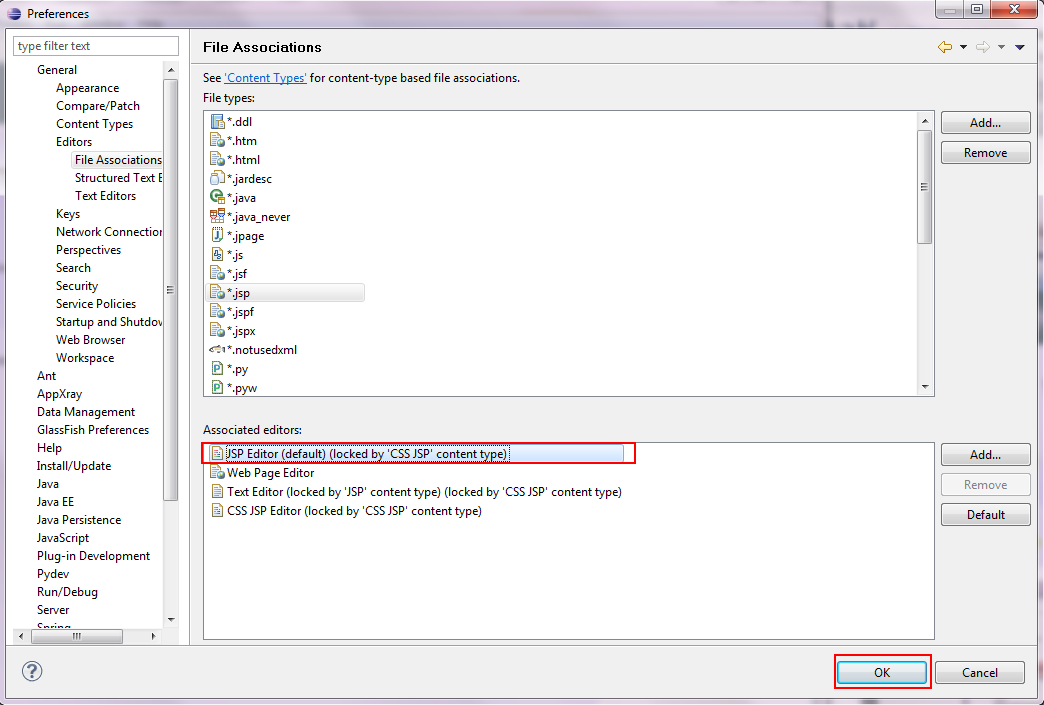
1. By default, jsp file will be opened using Web Page Editor which is not supported by most of the jsp. To resolve the issue, go **to Window > Preferences > General > Editors > File Associations** and change the default editor for jsp file to JSP Editor by select \*.jsp and JSP Editor then click ‘Default’.





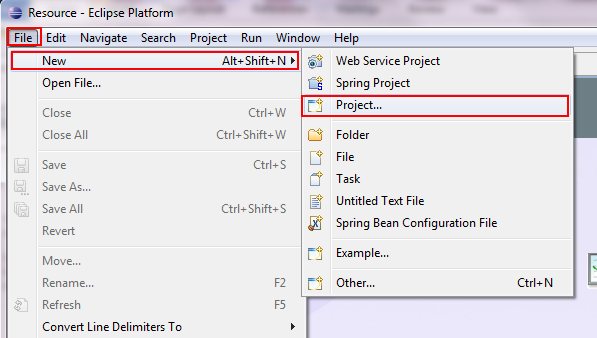
1. After you set JSP Editor as default editor for jsp file, you may find it located as the first selection in associated editors listing. Click ‘OK’ to activate such changes.

\*\*You may also customized default for other file extension according to your desired.

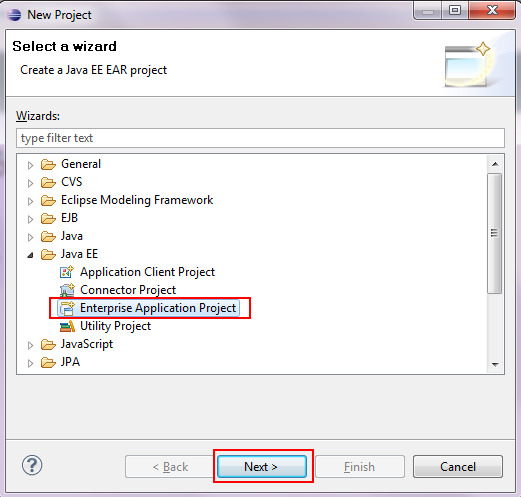


### Create AWS Enterprise Application Project

1. Next, create AWS project by go to **File > New > Project**.



1. Select **Java EE >** **Enterprise Application Project**. Click ‘Next’.

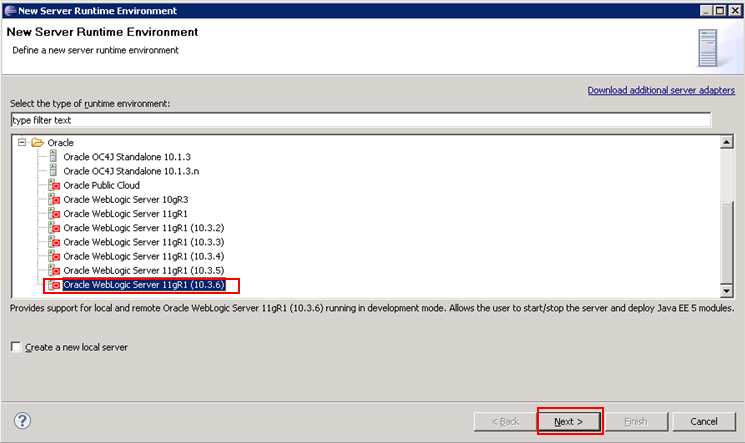


1. In new workspace, there will be no Target Runtime available. Kindly click on ‘New Runtime’ to create one for our project.

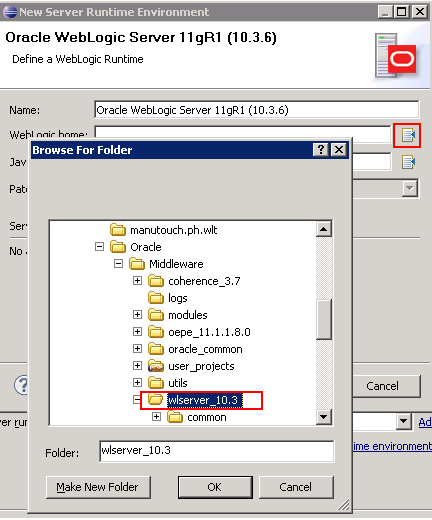
\*\*Only one time runtime creation is required for each new workspace.



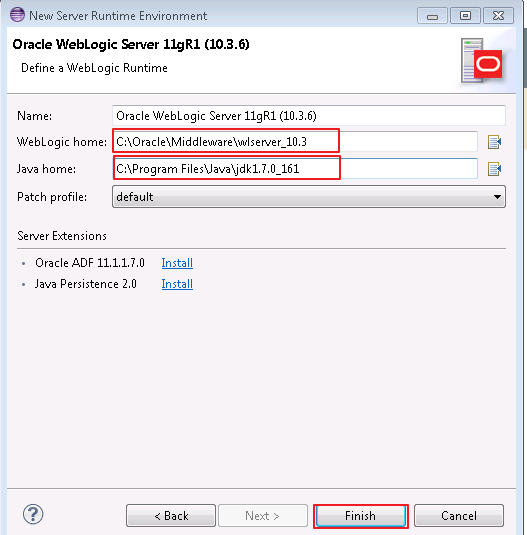
1. Choose Oracle Weblogic Server 11gR1 (10.3.6) under Oracle folder. Click ‘Next’.



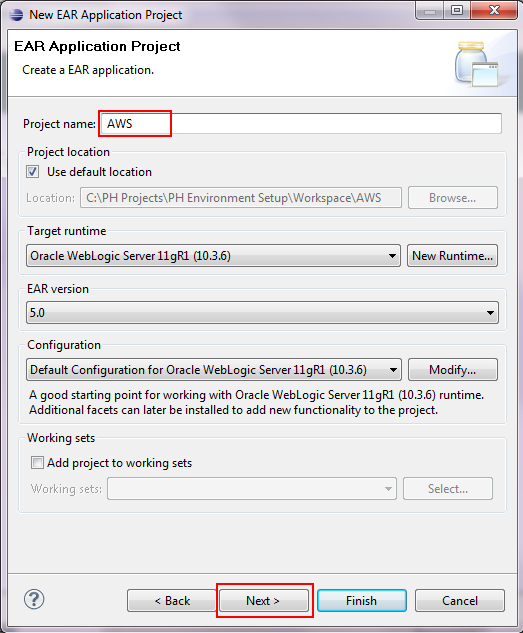
1. Choose  for Weblogic Home and direct to **<Oracle\_Home>\wlserver\_10.3**. Click ‘OK’



1. You may see Java Home have been automatically filled up upon choosing Weblogic home. Kindly confirm if your Java Home is pointed to **JDK1.7.0\_161**. Click ‘Finish’ to complete setup for runtime Environment.

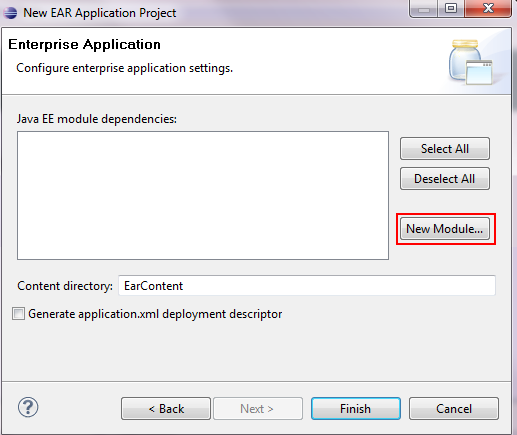


1. Target runtime, EAR version and Configuration will be auto populated upon completion of create new runtime. Kindly name the project as **AWS** and click ‘Next’.

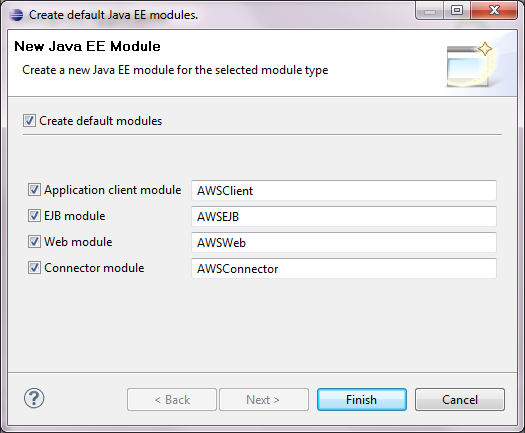


1. Next, you will be directed to create new module screen. We will need to create **EJB** and **WEB** module for **AWS** project. Click on ‘New Module’

\*\*Unlike AWS, we will only require to create WEB module for File Server

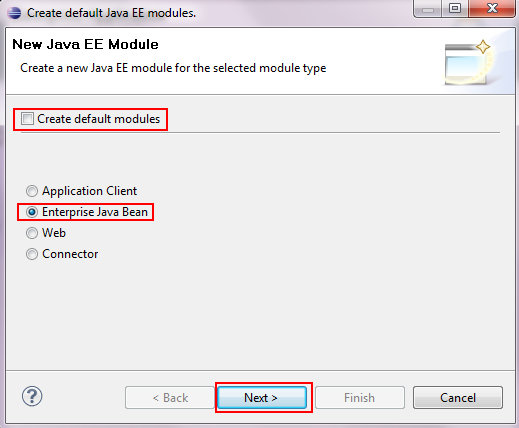


1. You will see figure as shown below.

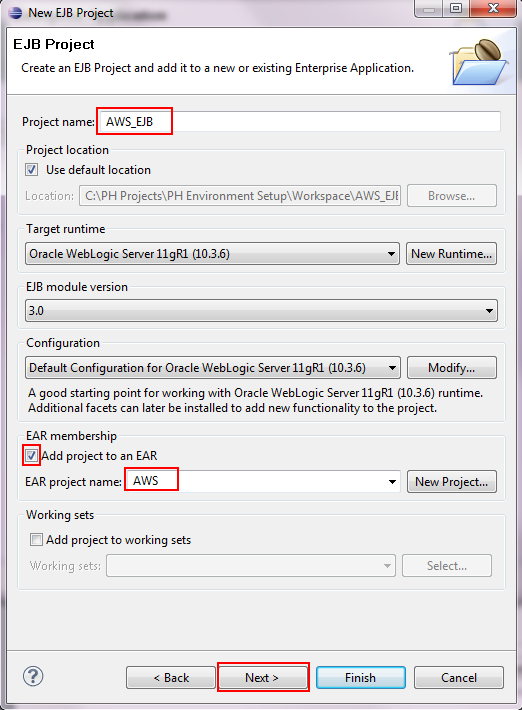


### Create EJB Module

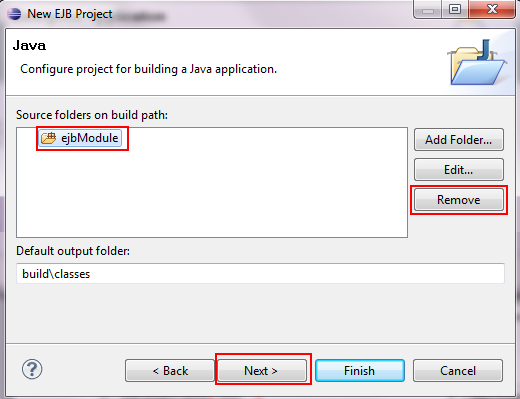
1. Uncheck **Create default modules** and select Enterprise Java Bean to create EJB Module. Click ‘Next’.



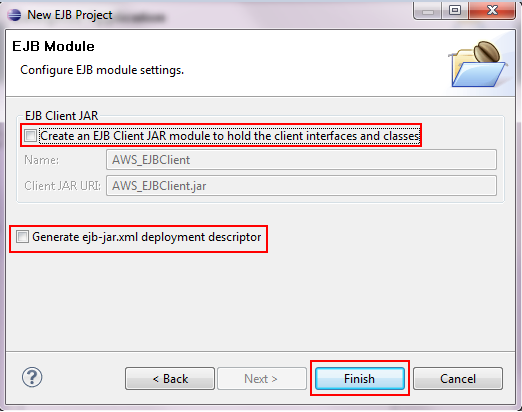
1. Another new project screen will be pop up, name the project as **AWS\_EJB**. Also, check on **Add project to an EAR** and select the **Enterprise Application Project** which created previously. Click ‘Next’.



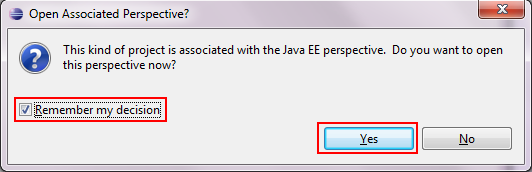
1. Ensure there is no source folder on build path. If there is any, kindly remove it by select the folder and click ‘Remove’. Click ‘Next’ upon finished.



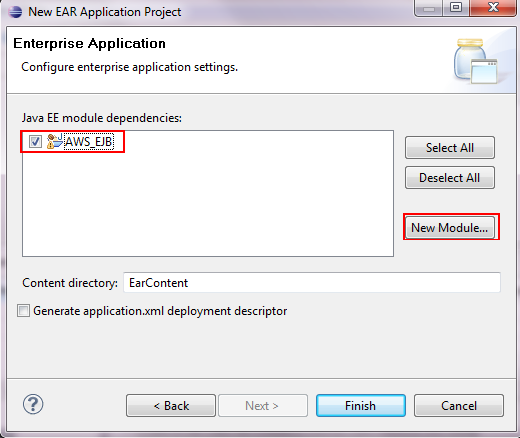
1. Ensure both **Create an EJB Client JAR module…** and **Generate ejb-jar.xml…** is uncheck. This is due to we will use back our existing ejb-jar.xml instead of creating new. Click ‘Finish’.



1. Upon finished, there might be a pop up asking if you want to open perspective view. Check on **Remember my decision** and click ‘Yes’.

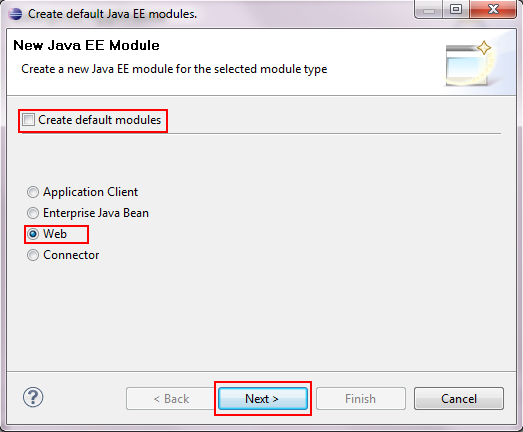


1. Next, you will be able to see the newly created **EJB** project shown on create module screen. Click on ‘New Module’ to create **WEB** project.

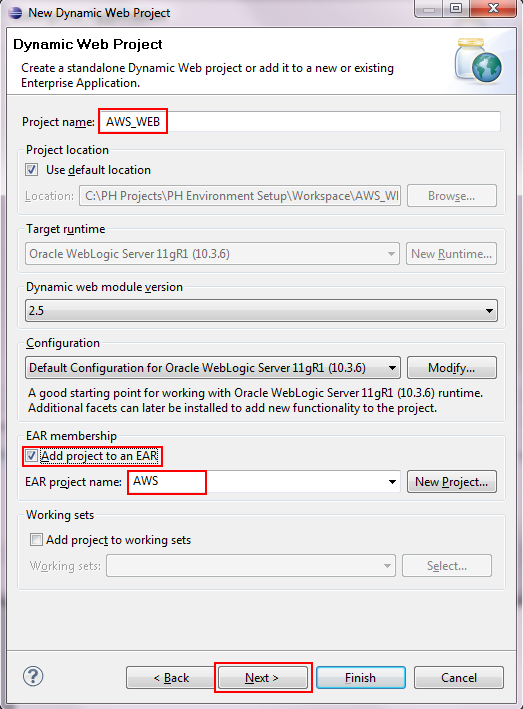


### Create WEB Module

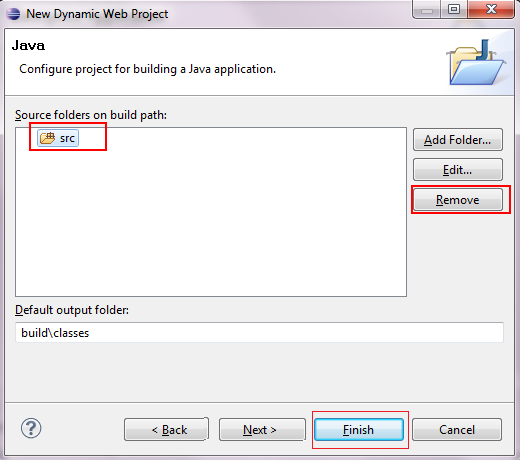
1. Similar with the way we create EJB module, however, we will be required to choose on **Web** and click ‘Next’.



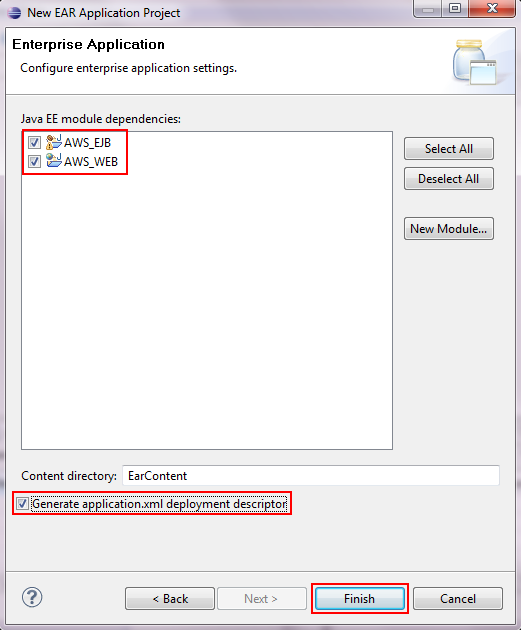
1. Another new project screen will pop up to create WEB project. Kindly named the project as **AWS\_WEB**. Also, check on **Add project to an EAR** and select the **Enterprise Application Project** which created previously. Click ‘Next’.



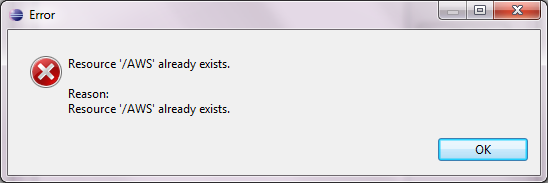
1. Ensure there is no source folder in build path. Remove if there is any and click ‘Finish.



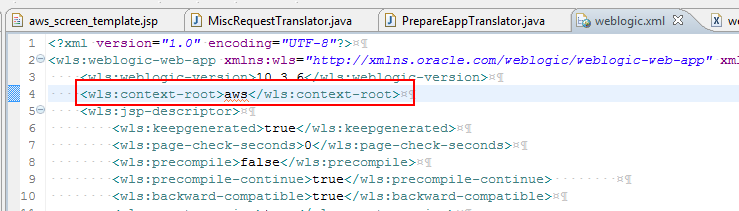
1. At create module screen, ensure both **AWS\_EJB** and **AWS\_WEB** project are checked. Also, check on **Generate application.xml …** and click ‘Finish’.



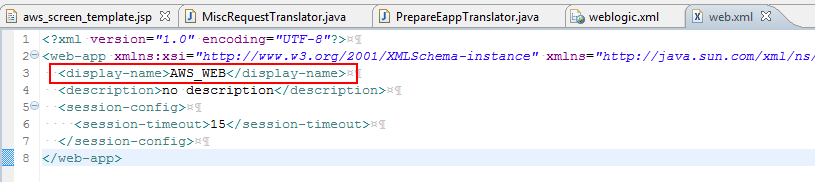
1. An error will prompt stated the **Enterprise Application Project** is already exists. Just click ‘OK’ and ignore the message.



1. Ensure that context-root is aws in weblogic.xml.

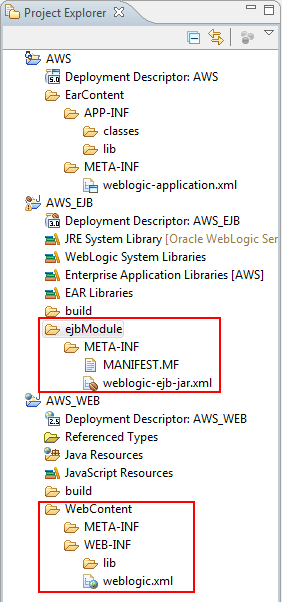


1. Ensure that display\_name is AWS\_WEB in web.xml

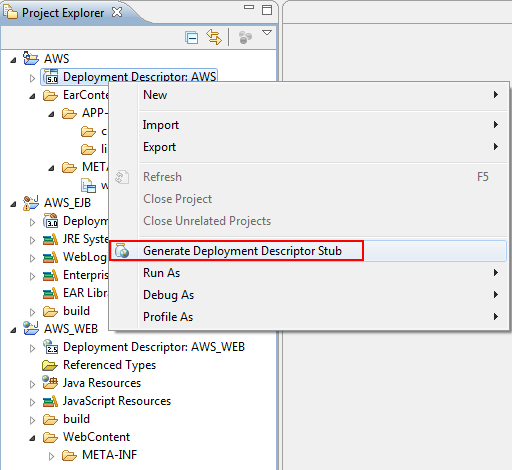


### Setup AWS Enterprise Application Project

1. Upon completion of create new **Enterprise Application Project**, you will find the 3 projects shown in your **Project Explorer**. Kindly remove the **ejbModule** folder from **AWS\_EJB** and remove **WebContent** from **AWS\_WEB** as we will reference those file in our source code.

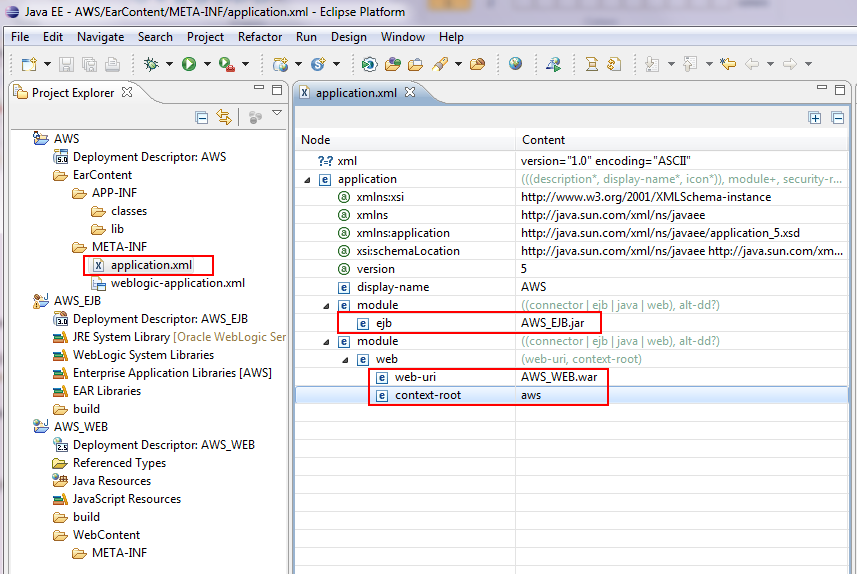


1. Next, right click on **Deployment Descriptor: AWS** and choose **Generate Deployment Descriptor Stub**.



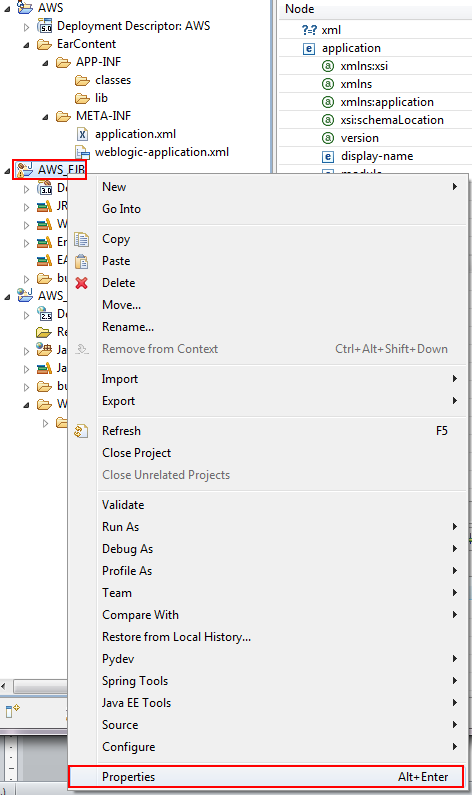
1. An application.xml will be generated. Open application.xml and ensure the AWS\_EJB.jar and AWS\_WEB.war is attached as a module. Also, ensure the context-root is point to aws.

\*\*Context-root for **File Server** will be fileserver.

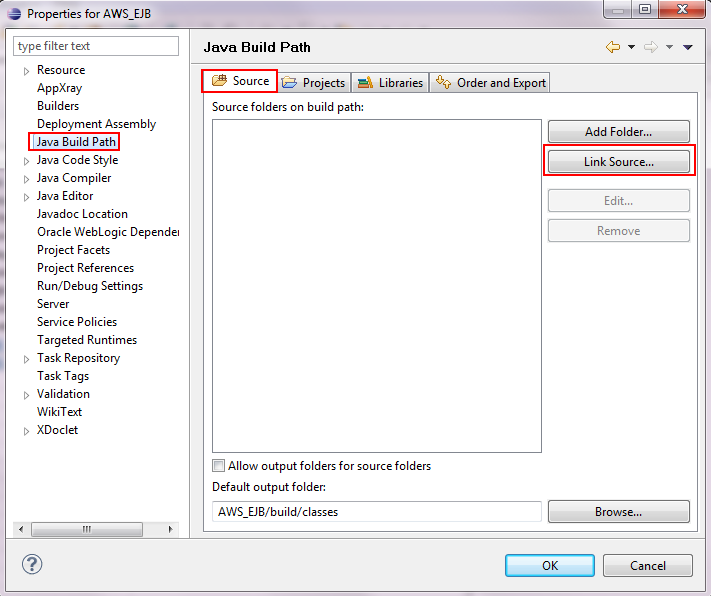


### Setup EJB Module

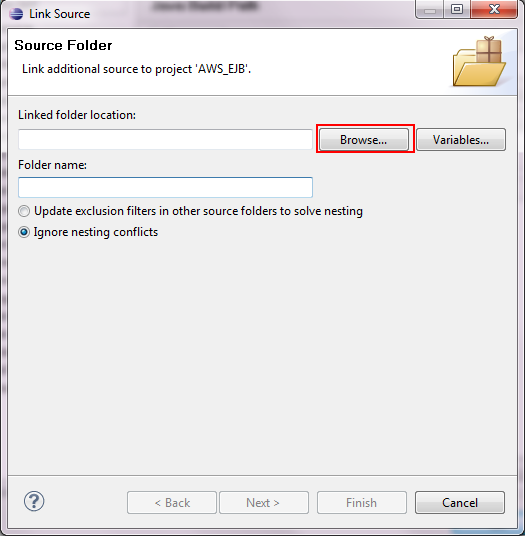
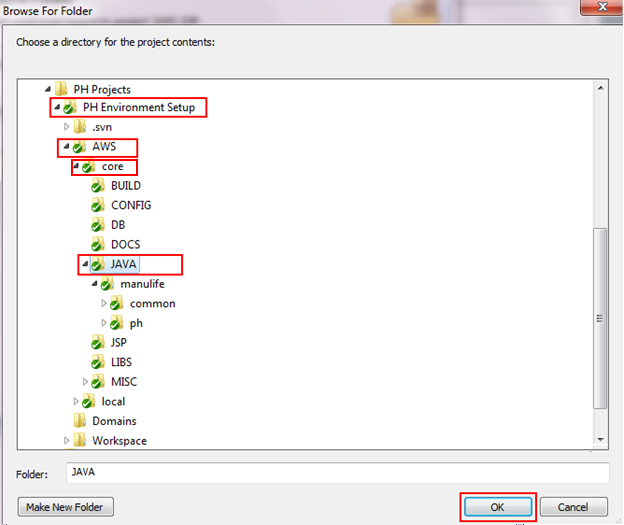
1. Followed, we need to link source for **AWS\_EJB** and configure the project setting. Right click on **AWS\_EJB > Properties**.



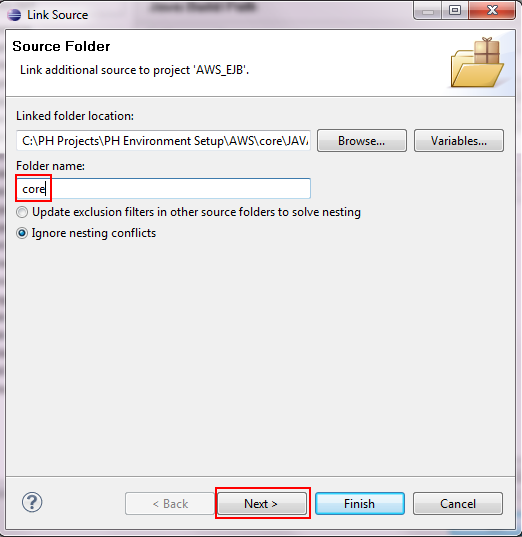
1. Go to **Java Build Path > Source** and click on ‘Link Source’



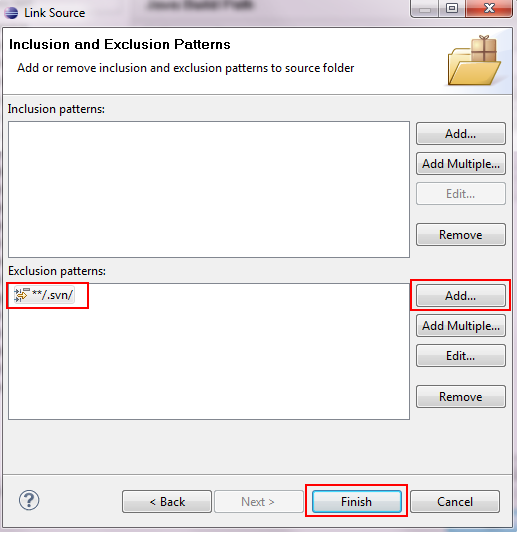
1. A pop up will be shown. Click on Browse and link to **aws\core\java**.

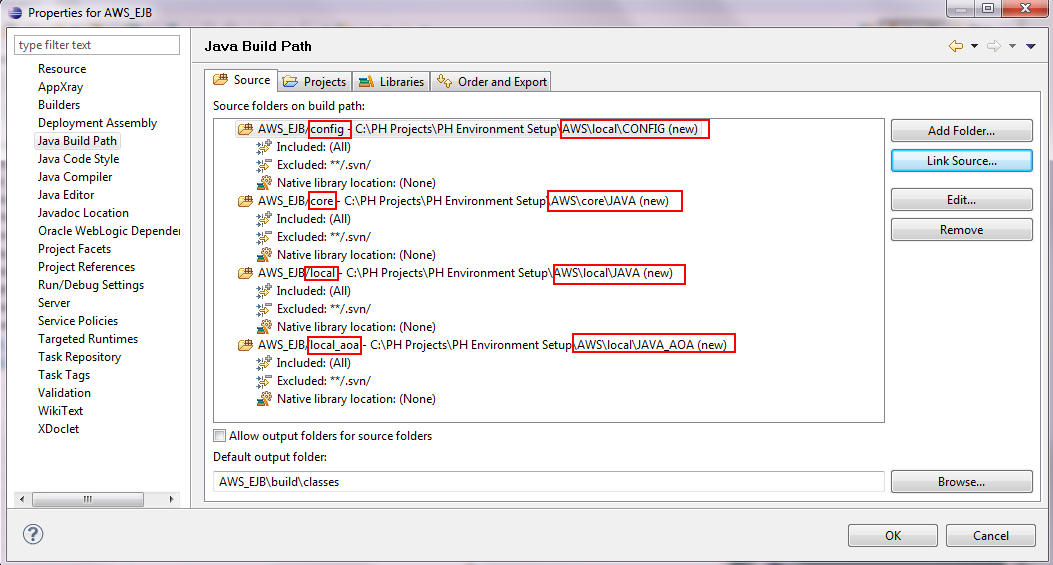
1. Name the folder name as core. Click ‘Next’.



1. Click ‘Add’ and included \*\*/.svn/ as the exclusion patterns. Click ‘Finish’

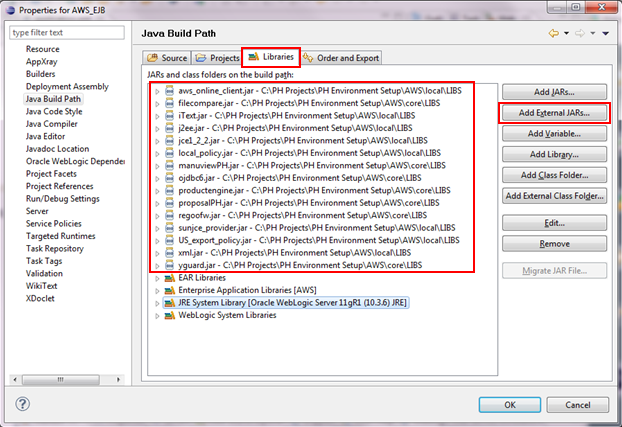


1. Repeat steps (ii) to steps (v) to link source for **aws\local\config**, **aws\local\java** and ***aws\local\java\_aoa\*(*** *Please skip the setup of* ***aws\local\java\_aoa)****.* Your link source screen should be similar with figure shown below.

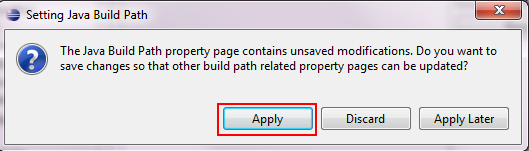


*\* Please skip the setup of* ***aws\local\java\_aoa****.*

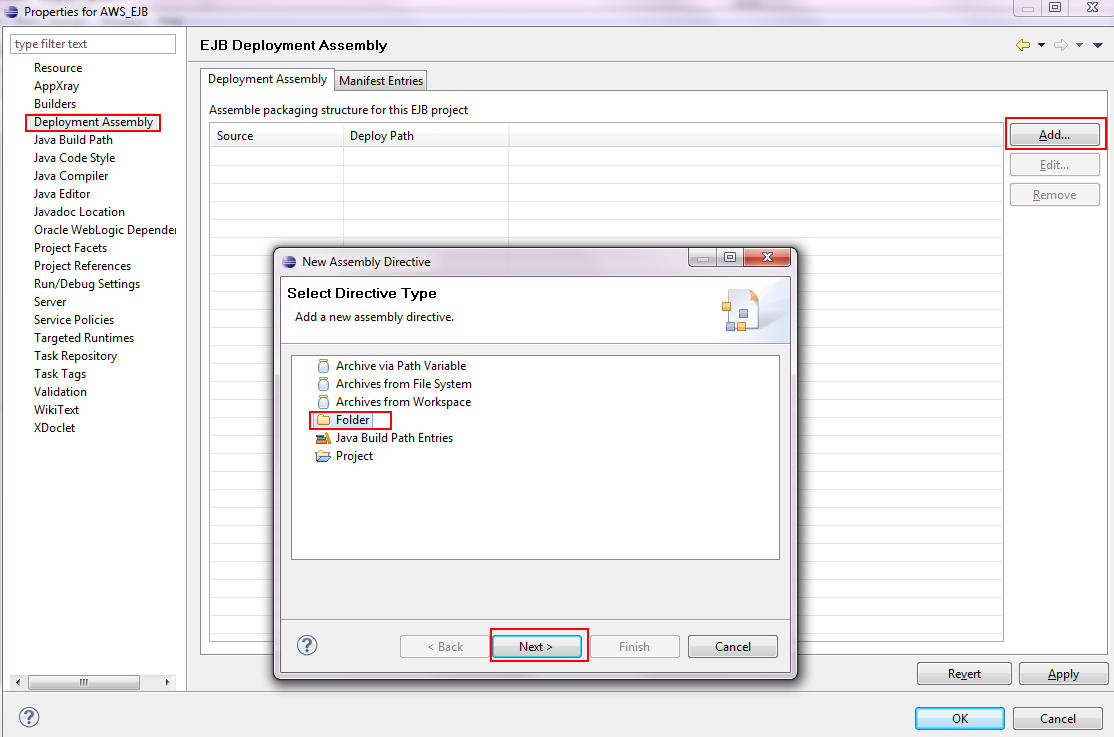
1. Next, go to **Libraries** and **Add External JARs** located in **aws\core\libs** and **aws\local\libs**. Below is all the libs should be included in **AWS\_EJB**.



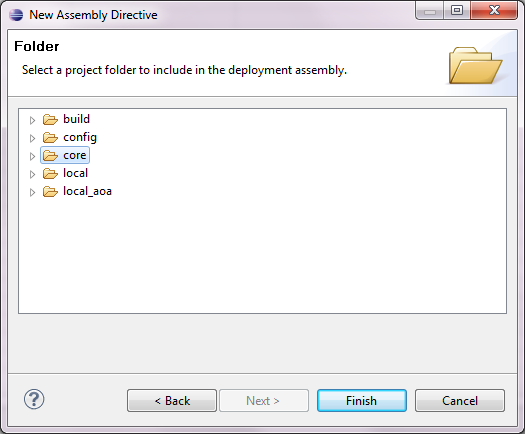
1. Next go to **Deployment Assembly** to configure the deploy path. You may be prompt whether to save your **Java Build Path** changes. Click ‘Apply’.



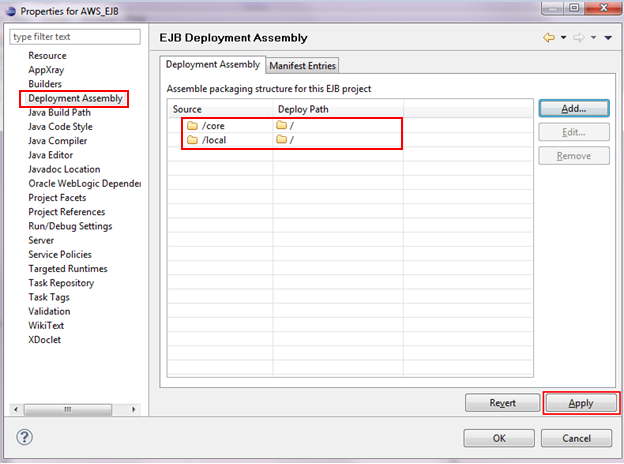
1. In **Deployment Assembly** click ‘Add’. Select ‘Folder’ and click ‘Next’.



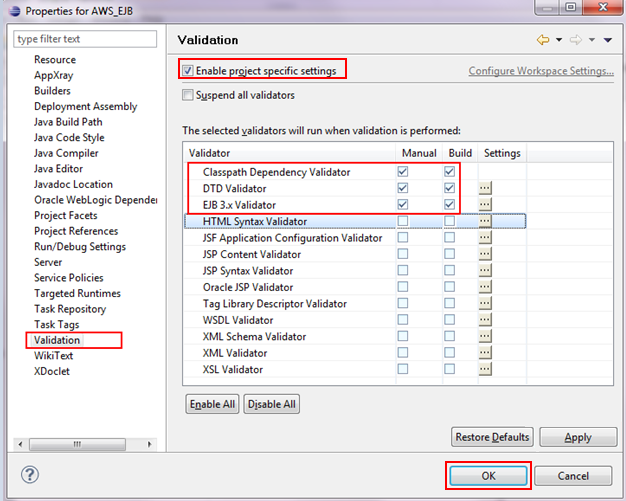
1. Choose **core** folder and click ‘Finish’.



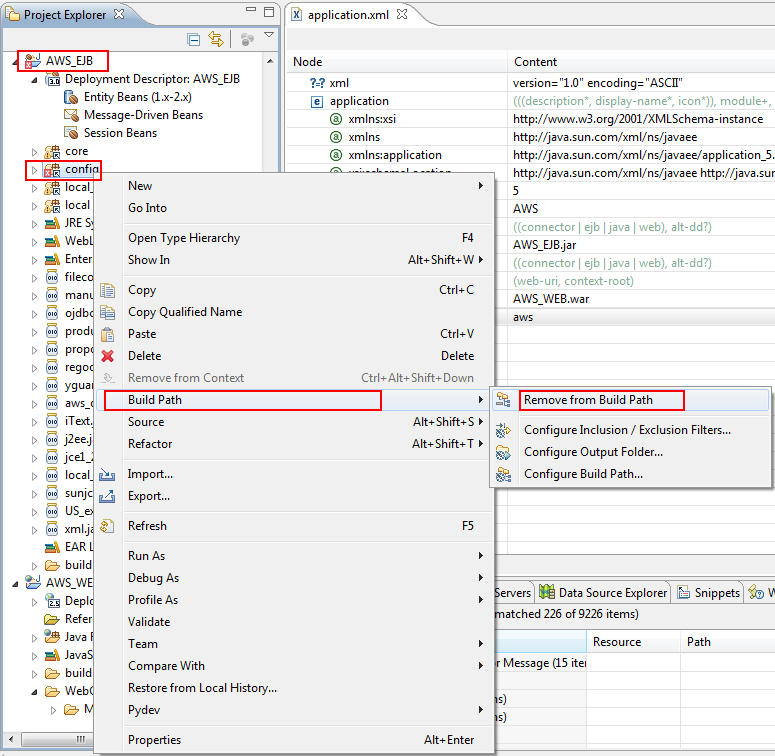
1. Repeat step (ix) and step (x) to make sure the deployment assembly is same as figure below. Click ‘Apply’.



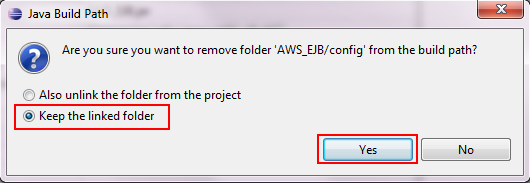
1. Next, go to **Validation**. Ensure **Enable project specific settings** is checked and only **Classpath Dependency Validator**, **DTD Validator** and **EJB 3.x Validator** is activated. Click ‘OK’.



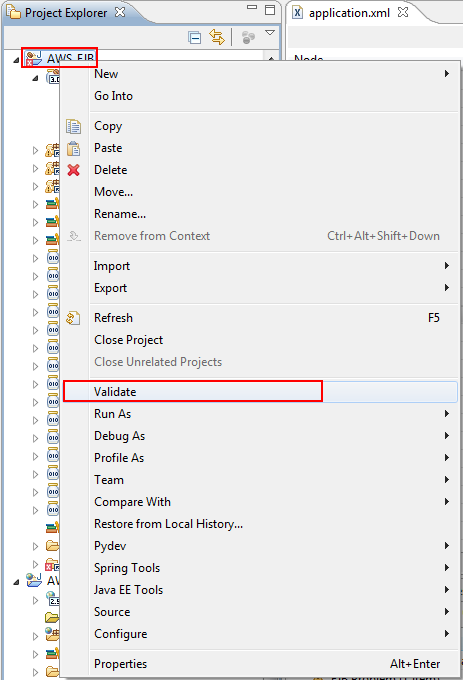
1. Back to Project Explorer, you will find there is error in **config** folder of **AWS\_EJB**. Kindly remove the folder from build path by **Right Click on config folder > Build Path > Remove from Build Path**.



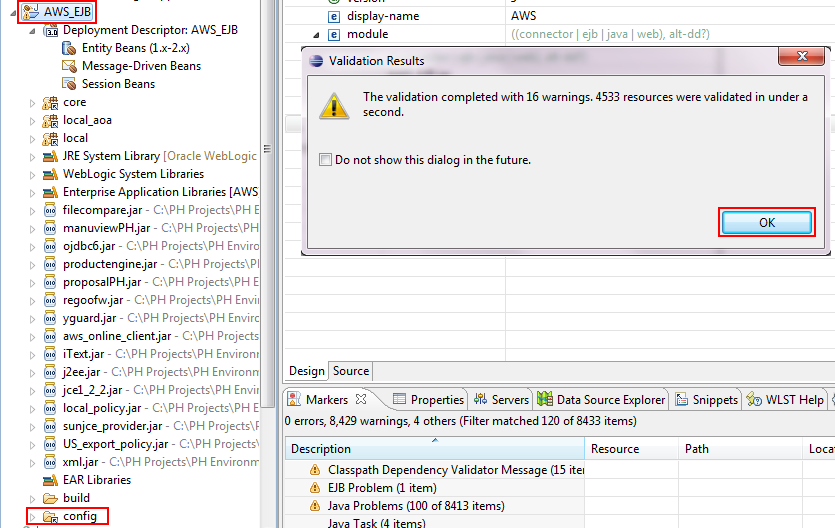
1. A pop up will to shown during remove the folder from build path. Select **Keep the linked folder** and click ‘Yes’.



1. Finally, validate **AWS\_EJB** project by Right Click on **AWS\_EJB > Validate**.

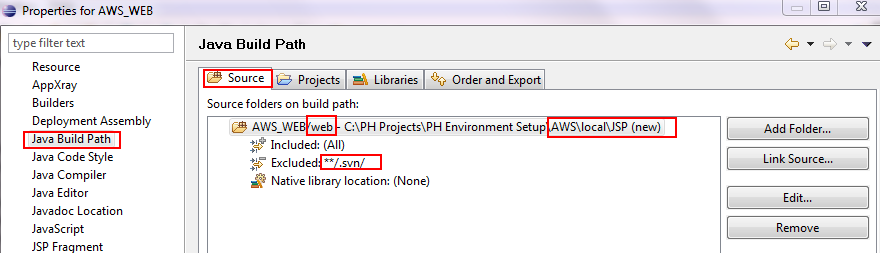


1. The project should be validated successfully with no errors shown.

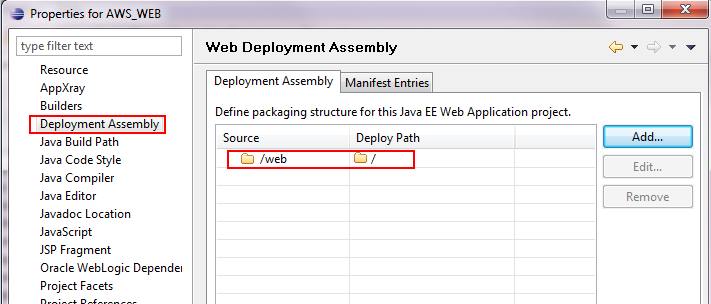


### Setup WEB Module

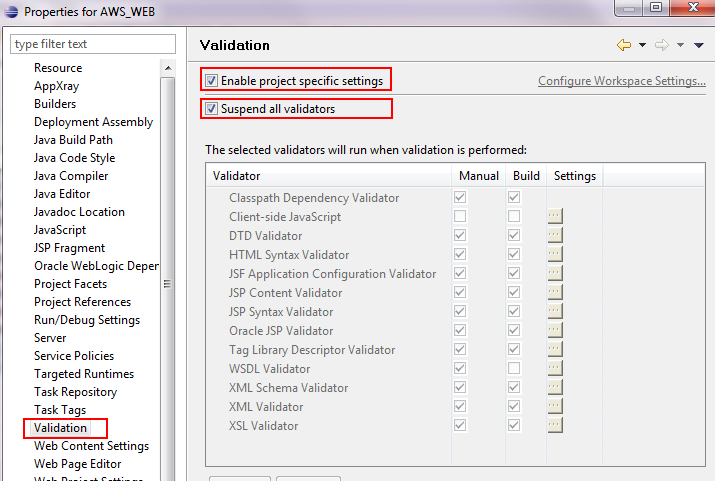
1. **Right Click AWS\_WEB > Properties > Java Build Path** and add link source to **aws\local\jsp**.



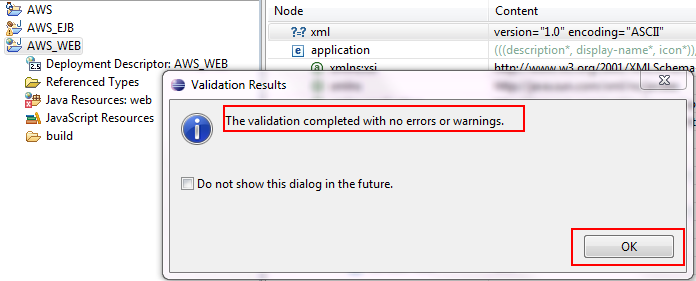
1. In Deployment Assembly, ensure only web folder is added.



1. Go to **Validation**, ensure **Enable project specific settings** and **Suspend all validators** is checked.



1. Finally, Right Click AWS\_WEB > Validate to validate AWS\_WEB project. You should find the validation is successful with no validation error.



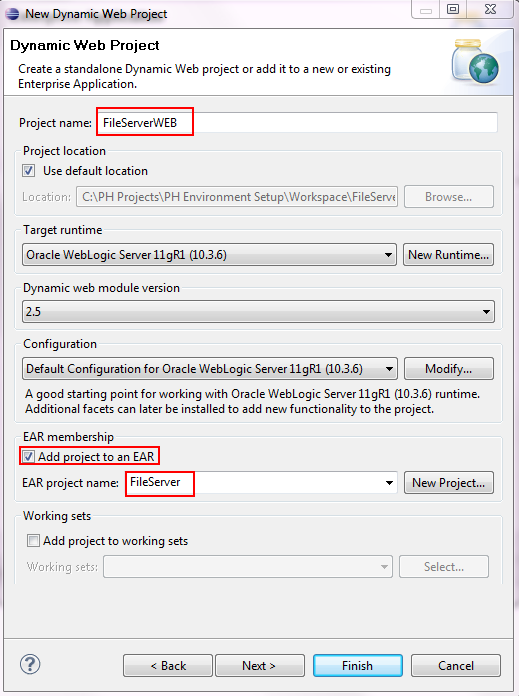
### *Notes: Setup for FileServer can be skipped for development environment setup as we will point the FileServer URL back to AWS.(For PH AWS system, just skip the item and go to item ‘*Create Weblogic Server*’)*

### Create and Setup for File Server Enterprise Application Project with WEB module

Kindly repeat steps in [Create AWS Enterprise Application Project](#_Create_AWS_Enterprise), [Create WEB Module](#_Create_WEB_Module), [Create WEB Module](#_Create_WEB_Module) and [Setup WEB Module](#_Setup_WEB_Module).

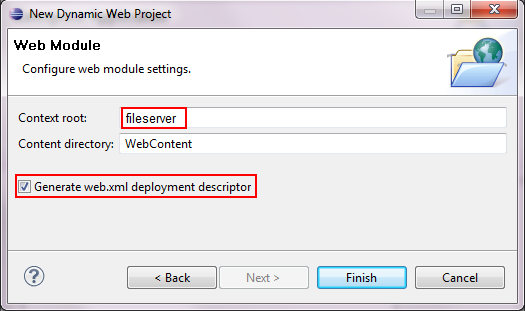
The only difference between AWS and File Server setup is:-

1. Named Enterprise Application Project as **FileServer**.
2. Name WEB Project as **FileServerWEB**.
3. Ensure FileServerWEB is added to FileServer EAR project.

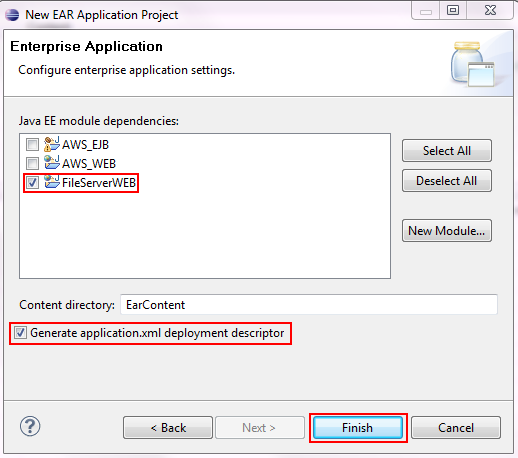


1. At [Create WEB Module](#_Create_WEB_Module) step (xix), click ‘Next’ instead of ‘Finish’.

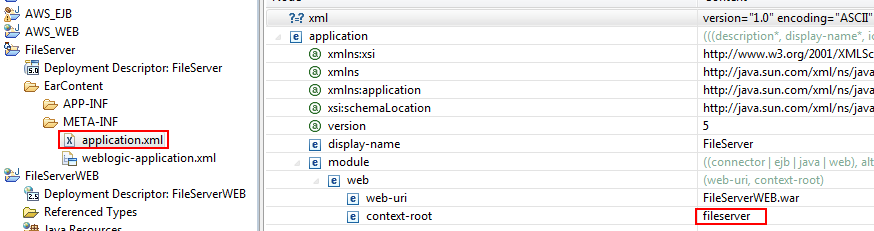
Ensure context root is set to **fileserver** and **Generate web.xml deployment descriptor** is being checked.



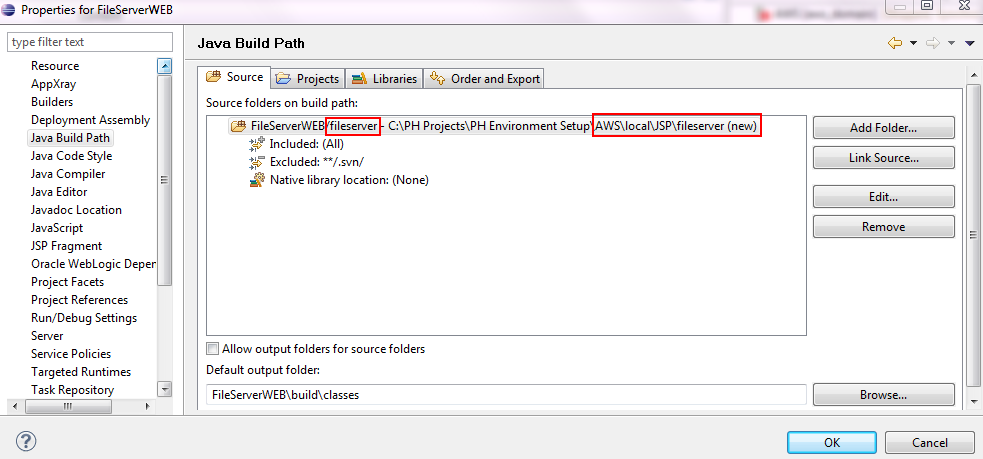
1. Ensure only **FileServerWEB** module is selected in create new module screen and **Generate application.xml deployment descriptor** is being checked.



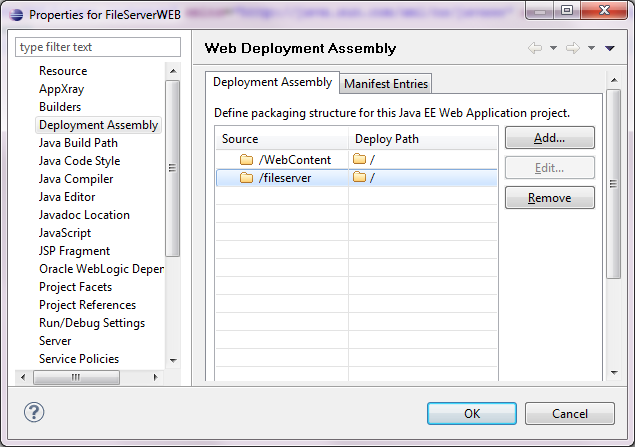
1. Ensure context-root for **application.xml** in **FileServer** Project is set to fileserver.



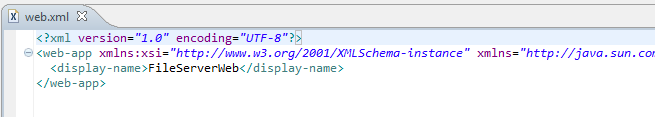
1. Ensure **aws\local\jsp\fileserver** is being to to AWS\_WEB project.



1. Ensure Deployment Assembly structure as shown in figure below.

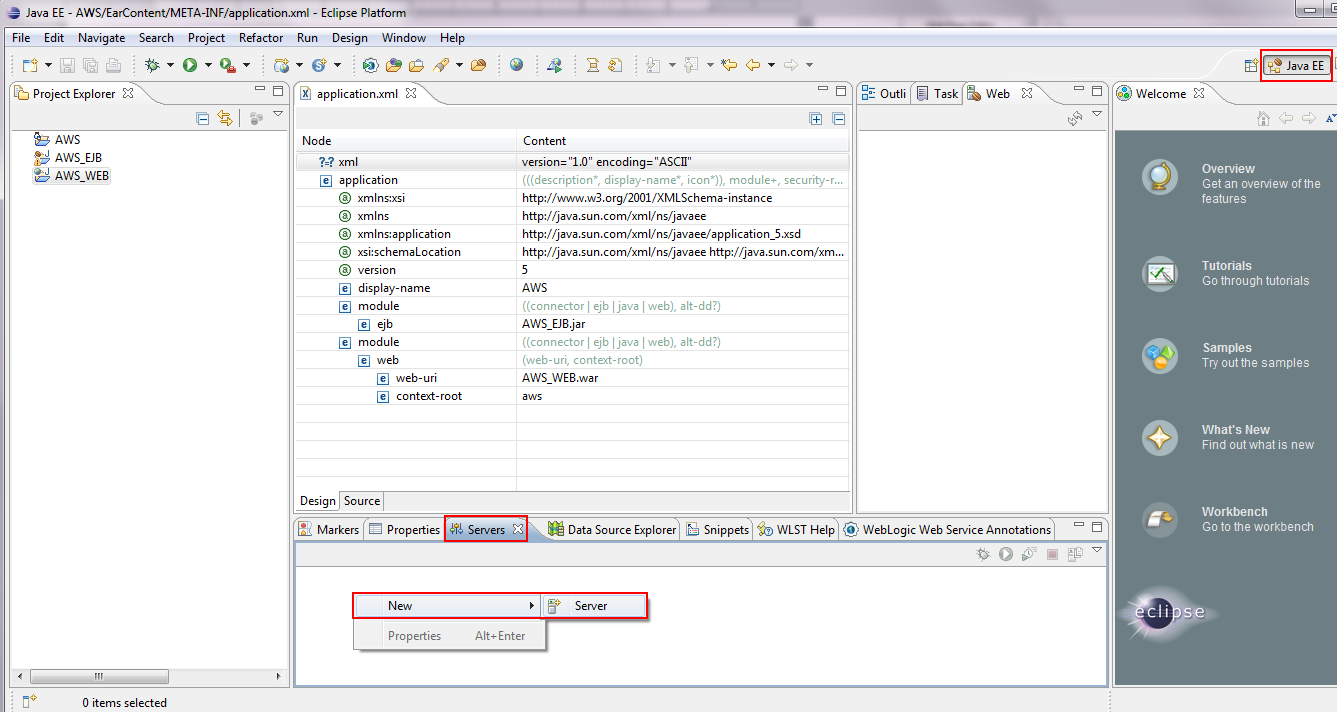


1. Ensure the content of web.xml in FileServerWEB is shown as below.

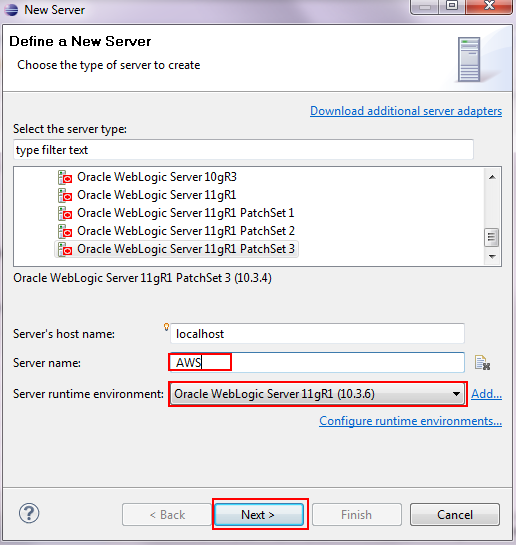


### Create Weblogic Server

1. Ensure you are in **Java EE** perspective, click on **Server** tab at the bottom of your workspace. Click new server by right click on the empty space in Server tab then **New > Server**.

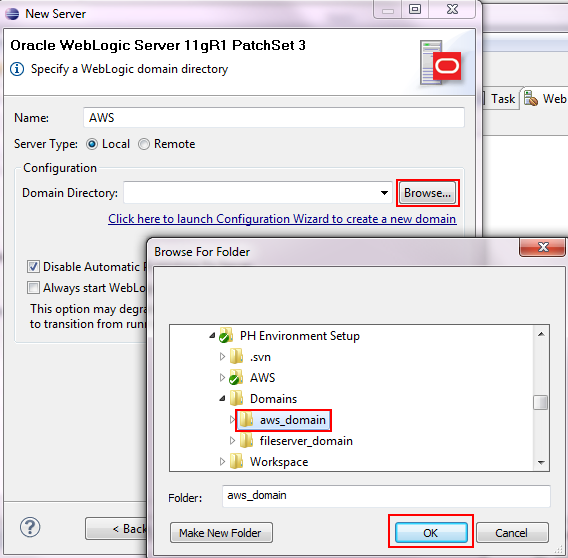


1. A New Server pop up will be shown. Enter name for your server and ensure your server runtime environment is select correctly. Click ‘Next’.

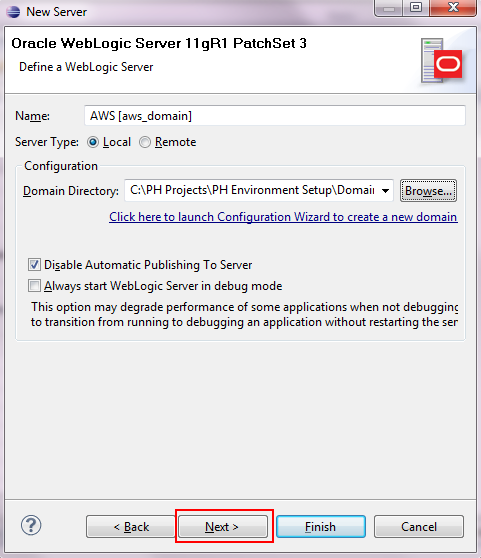


1. Click on ‘Browse’ and direct to path where you installed aws domain previously. Click ‘OK’.

\*\*For File Server, direct to file server domain which created previously

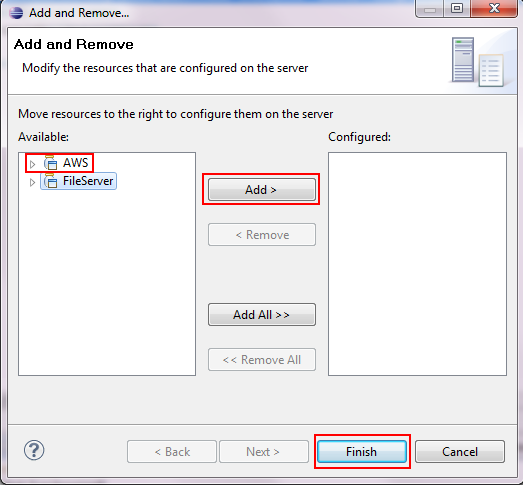


1. Ensure your weblogic server is defined as figure below. Click ‘Next’.



1. Followed, select **AWS** project and click ‘Add >’. Click ‘Finish’.

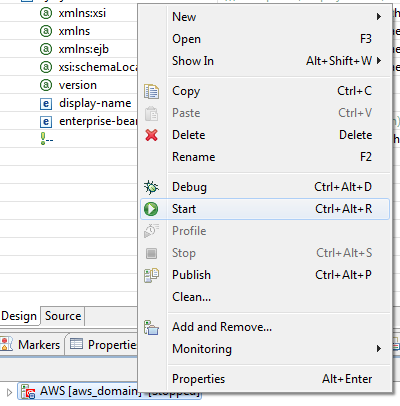
\*\*For File Server, select **FileServer** instead of **AWS**.



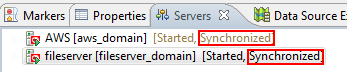
1. You may find the server being created and shown under Server tab.

To startup server, right click on the server and choose **Start**.

\*\* If you wish to debug your program, you may choose **Debug** instead of Start.

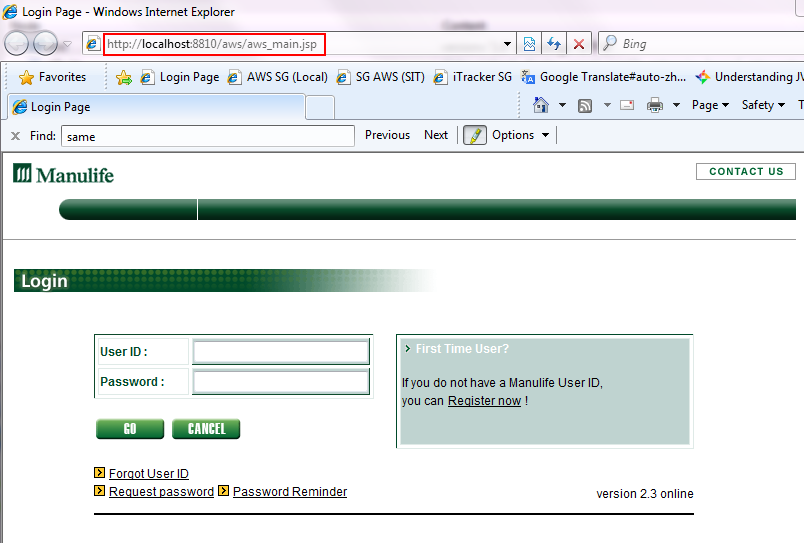


1. Your server should be shown Synchronized if there is no problem startup the server.



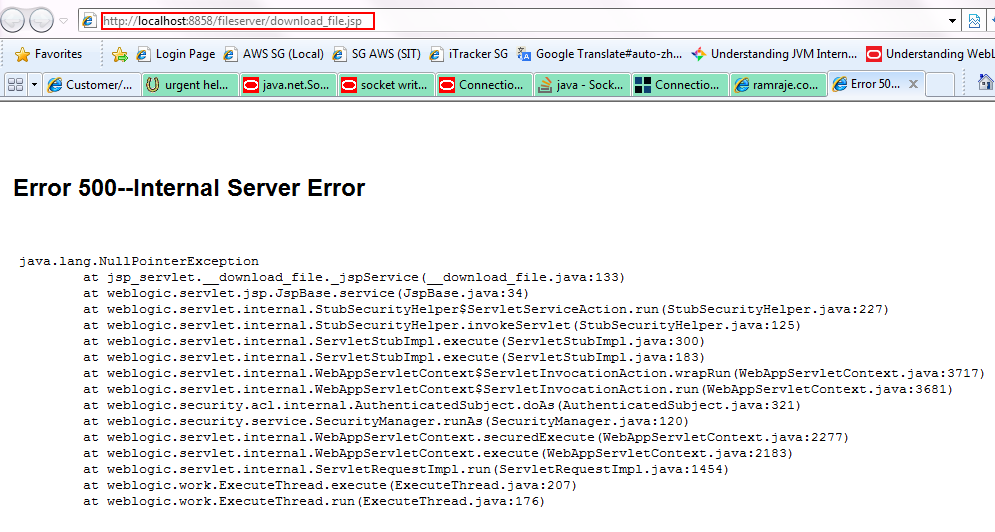
1. You may now launch your website by enter the url shown as below.

\*\*The port number should follow the port number you set during domain creation. In your case, your application is deploy to AdminServer of the domain.



1. Repeat step (i) to (vii) for **FileServer** server creation. Enter url below to see if you able to connect to fileserver. You may find error 500 on the webpage due to the path if not meant for us to connect in such way. However, this is proven your fileserver is ready to use.

\*\*The port number should follow the port number you set during domain creation. In your case, your application is deploy to AdminServer of the domain.

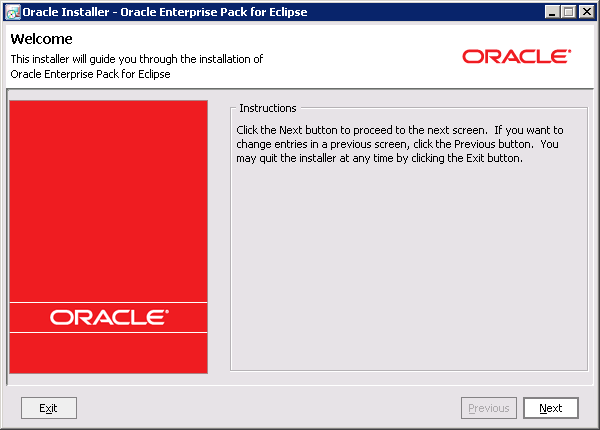


## Appendix A - Setup Weblogic and OEPE

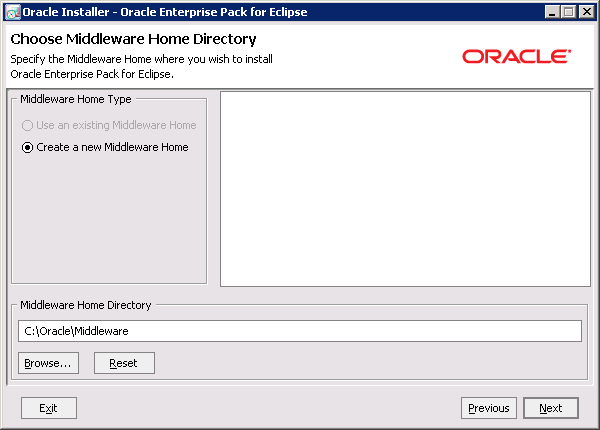
### Installing OEPE

(Can skip this part if already have OEPE and Weblogic in your environment, jump to next section)

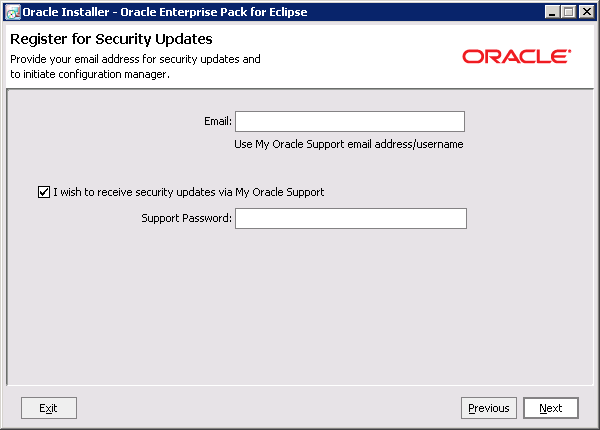
1. **Start** installer “**oepe-wls-indigo-installer-11.1.1.8.0.201110211138-10.3.6-win32.exe” (Ask your supervisor about the installer)**
2. Click **Next** to continue



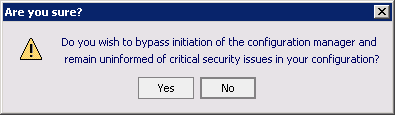
1. At the Choose **Middleware Home Directory** screen, select

**“Create a new Middleware Home** then click Next. 

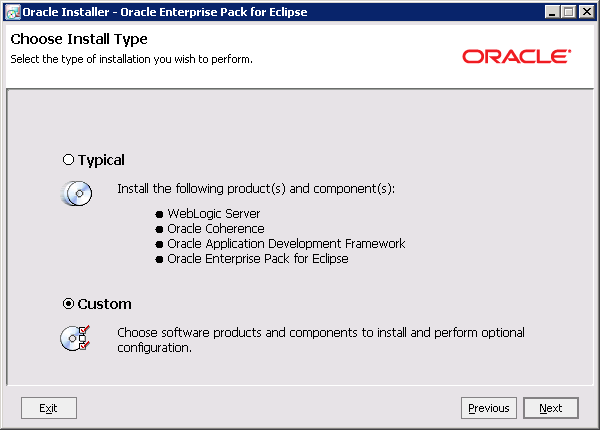
1. At the **Register for Security Updates** screen, **uncheck** the checkbox “**I wish to receive security updates via My Oracle Support**”.



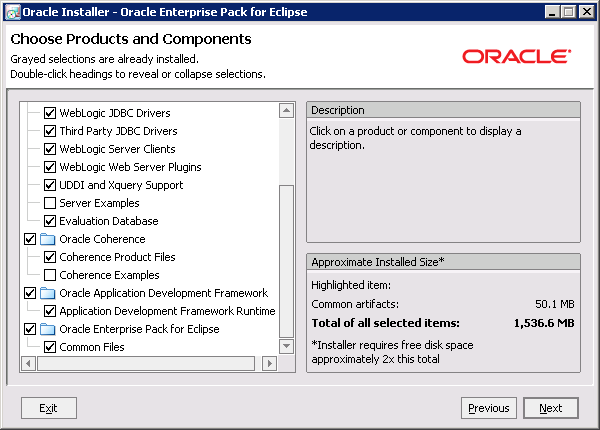
1. A warning will be shown, select **Yes**. Once done, click on **Next**.



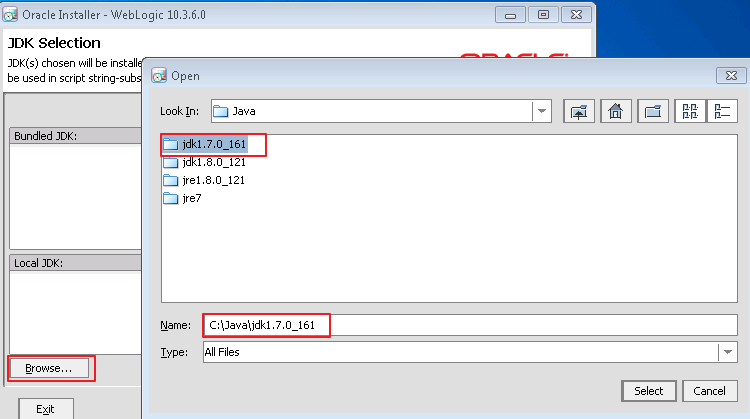
1. At the **Choose Install Type** screen. Select “**Custom**” and click on **Next**.

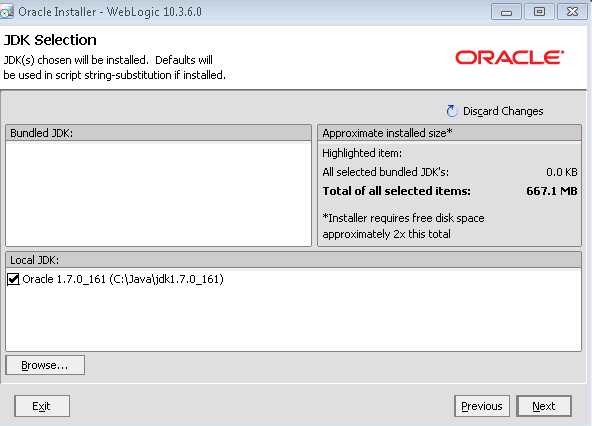


1. At the **Choose Products and Components** screen, leave everything as its defaults. Once done, click on **Next**.

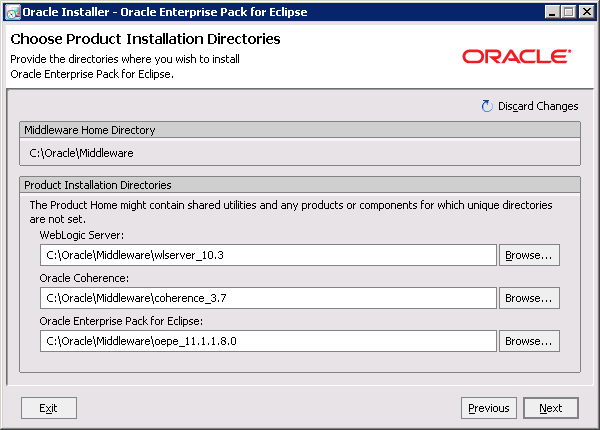


1. At the **JDK Selection** screen, click browse and point to location which you installed JDK 1.7.0.161 previously. Once done, click on **Next**.

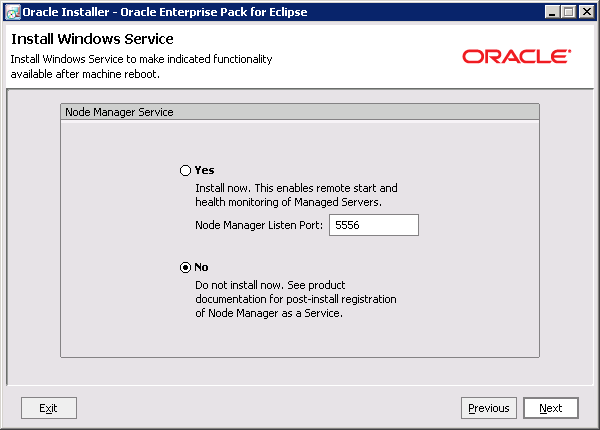




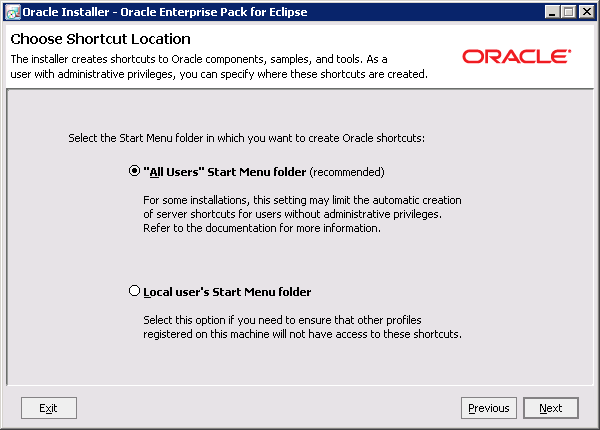
1. At the **Choose Product Installation Directories** screen, ensure that the installation home directory is “**C:\Oracle\Middleware**”. Once done, click on **Next**.



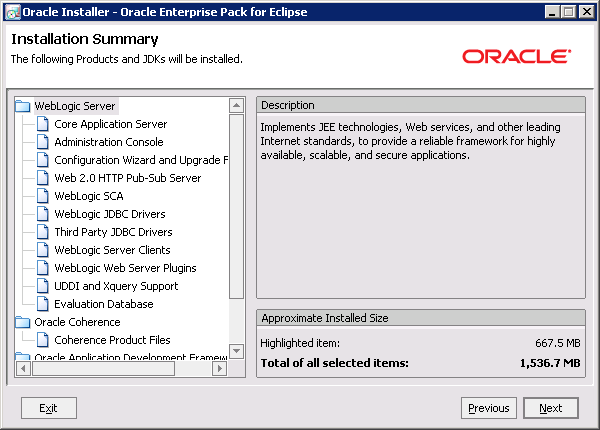
1. At the **Install Windows Service** screen, leave everything as its defaults. Once done, click on **Next**.



1. At the **Choose Shortcut Location** screen, leave everything as its defaults. Once done, click on **Next**.



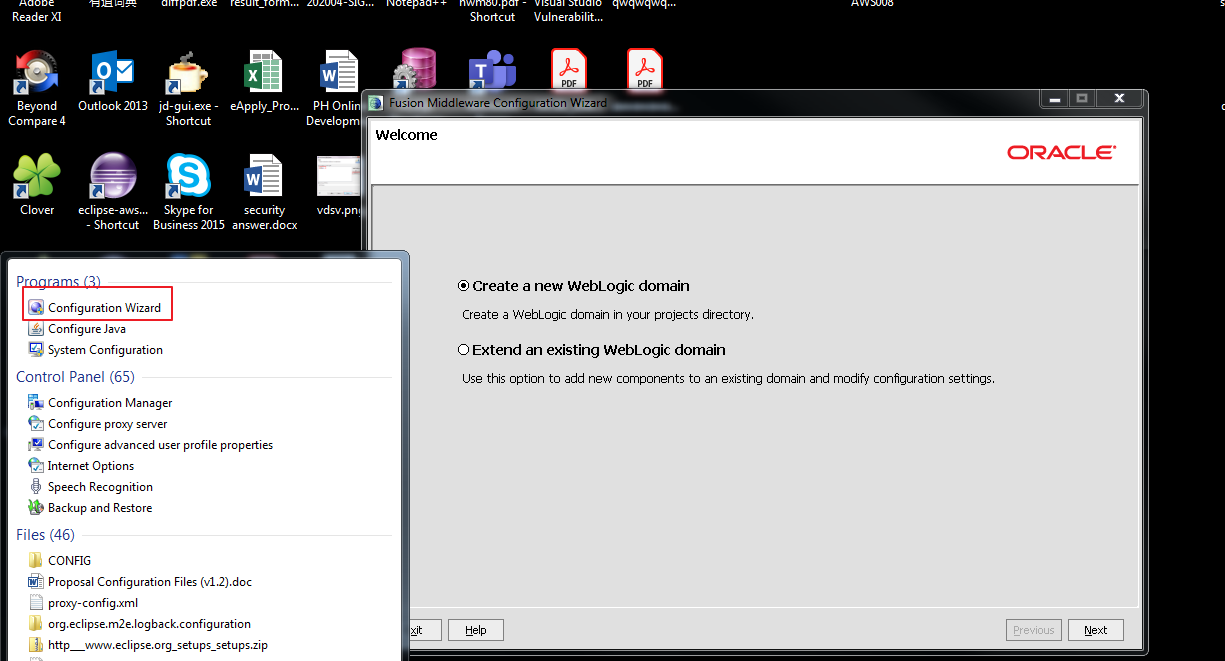
1. The above **Installation Summary** screen will be shown. Click on **Next** to start installing Oracle Weblogic.



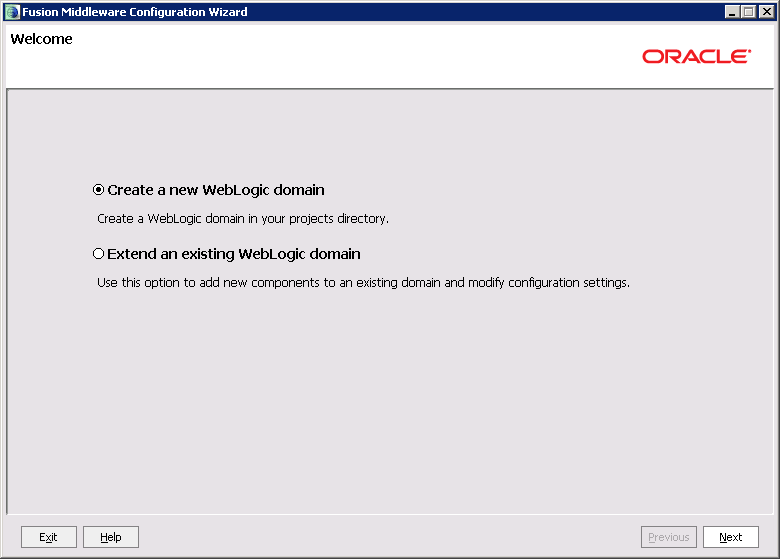
1. Uncheck “**startup Quickstart**” and click “**Done**” upon finish.

## Appendix B – Domain Creation

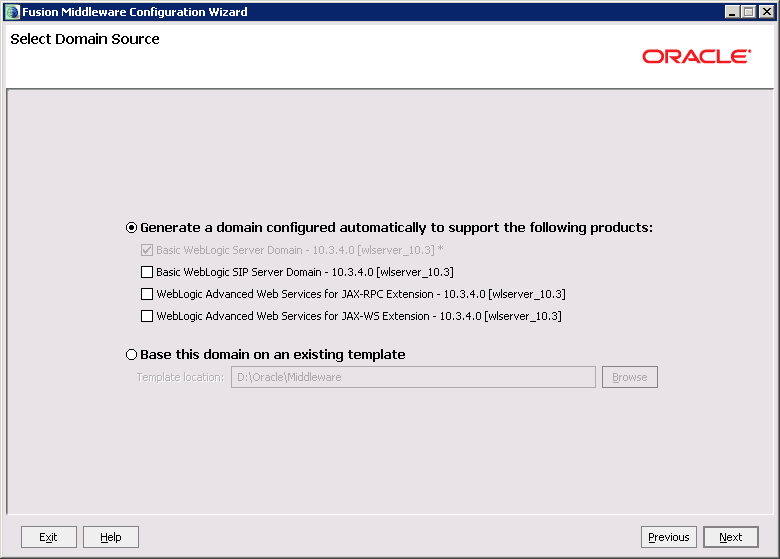
If it do not pop-up the next screen, you can also find it in windows button;



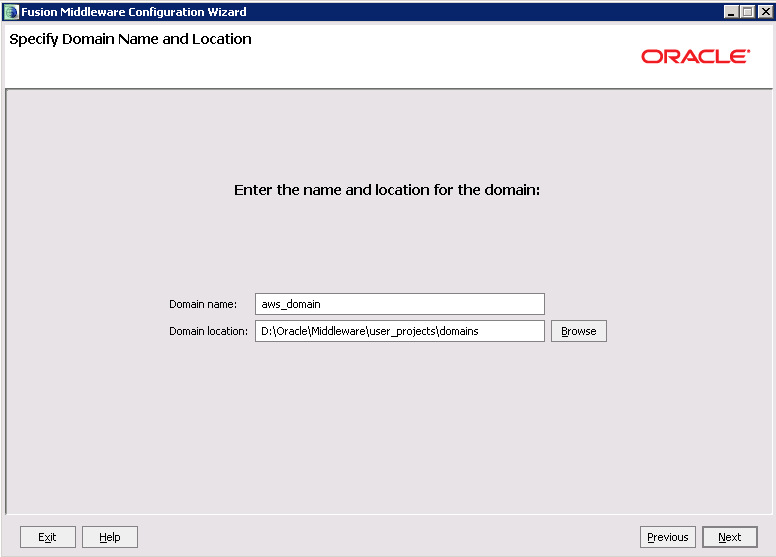
1. Select “Create a new WebLogic domain”, click “Next”.



1. Select the default option show in the picture and click “Next”.



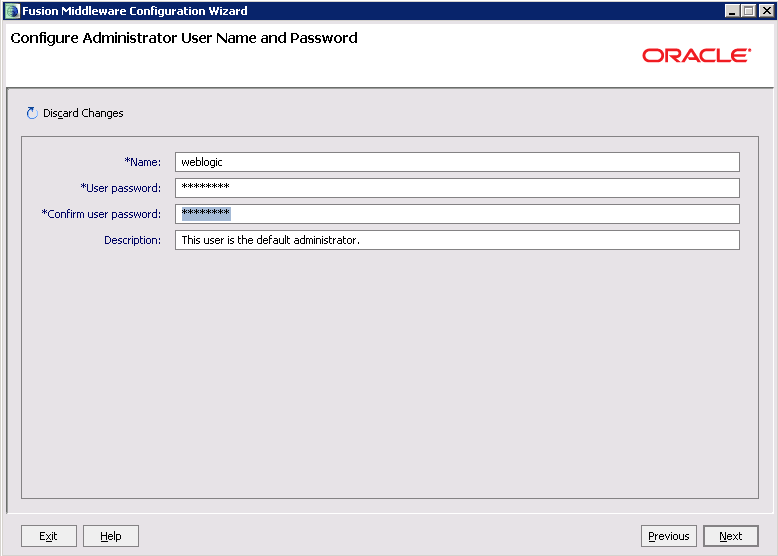
1. Key in Domain Name – “aws\_domain” / “fileserver\_server”. Use the default Domain location. Click “Next”.( fileserver\_server is not needed for PH AWS)



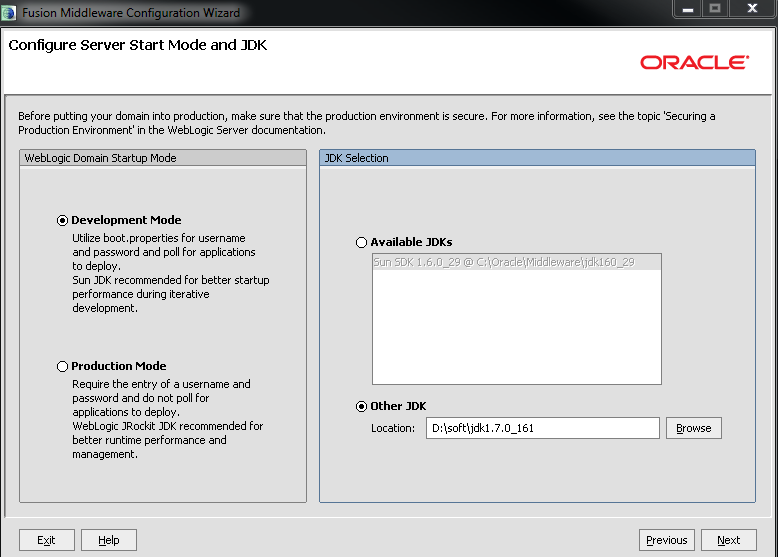
1. Use the default Name (Admin ID) as “weblogic”, then key in the desired password. Click “Next”.

\*\*The password will be used to logon to weblogic console.

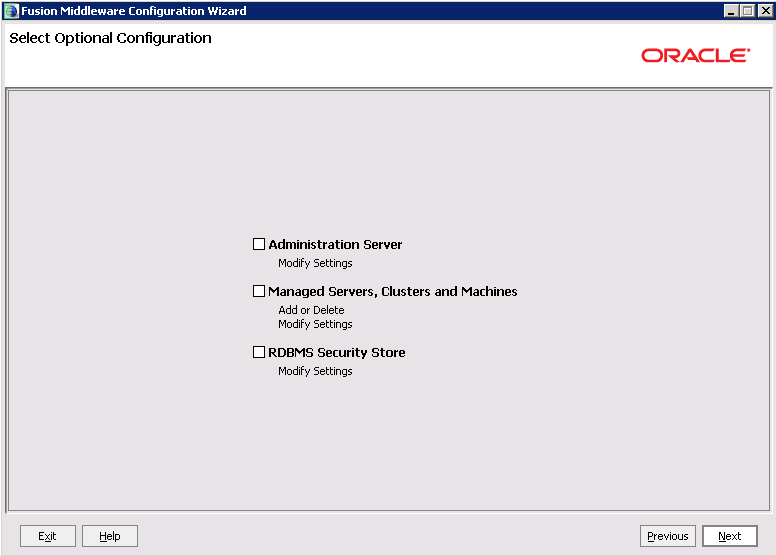
passw0rd



1. Select “Development Mode”, ensure you JDK selected is **JDK1.7.0\_161**. Click “Next”.



1. By default, the listening port for admin server will be 7001. If you would like to point to other port, kindly checked on administrator server and change the port number. Else, select nothing, click “Next”. (for AWS just as default)



1. Click “Create”.



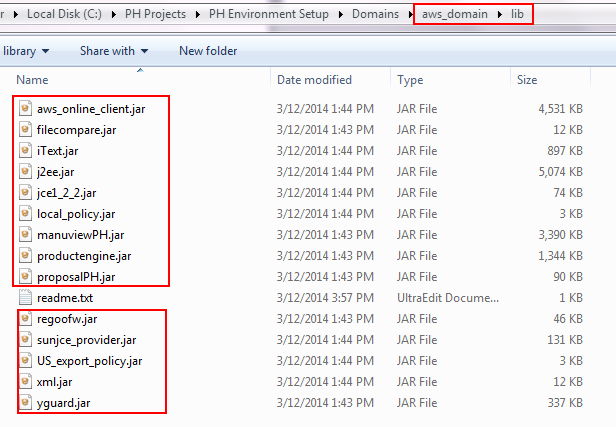
1. Wait for process finish, then click “Done”.

## Appendix C – Domain Setup

### AWS Domain

1. Copy all lib from **aws\core\lib** and **aws\local\lib** into **<domain>\lib**.

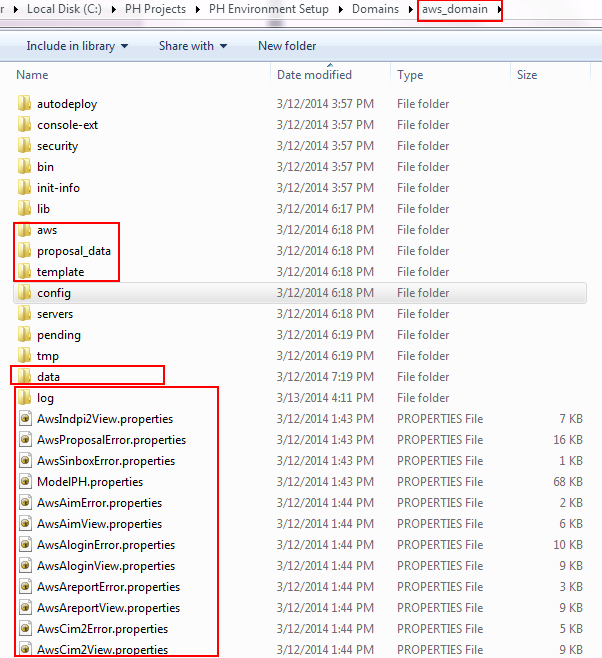
\*\*Except ojdbc6.jar



1. Copy all files located in **aws\local\config** into **<domain>**.

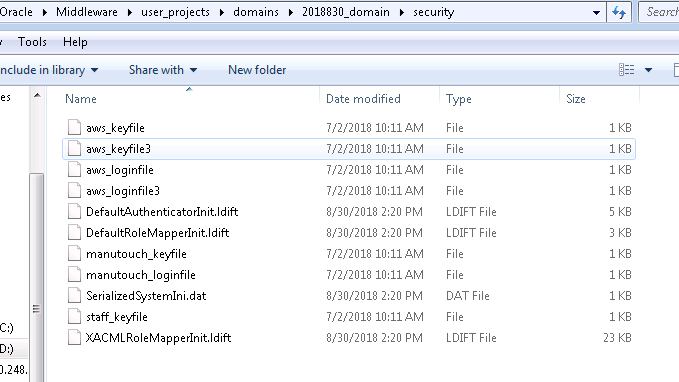
\*\*Except .xls files and properties\_production files.

1. Copy data folder which contains all catalogs and schema into **<domain>.**
2. Create empty **log** folder in **<domain>. In the AwsResources\_en\_PH.properties, please change the right data base username and password.**



1. Copy the files which in the red box into your domain **security** folder.



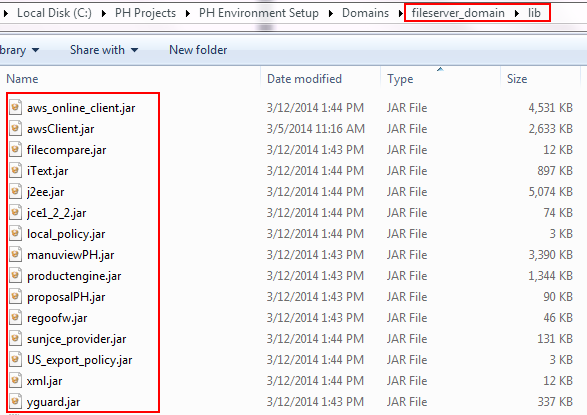


’

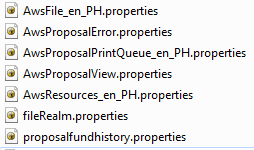
### File Server Domain(For PH AWS system this is not needed)

1. Copy all lib from **aws\core\lib**, **aws\local\lib** and **awsClient.jar** into **<domain>\lib**.

\*\*Except ojdbc6.jar.



1. Copy **aws\local\config\proposal\_data** (except .xls files) into **<domain>**.
2. In **aws\local\config**, copies properties files as listed below into **<domain>**.



1. Copy data folder which contains all catalogs and schema into **<domain>.**
2. Create empty **log** folder in **<domain>**.

