



# **Application Notes for Packaging and Deploying Avaya Communications Process Manager Sample SDK Web Application on BEA Weblogic Application Server Using BEA Workshop – Issue 1.0**

## **Abstract**

These Application Notes describe the steps required to package and deploy Avaya Communications Process Manager (CPM) sample SDK web application on a BEA Weblogic Application Server using BEA Workshop and the BEA Weblogic administrative console.

The BEA Weblogic Application Server is an open integration and application platform. It supports Java technology and provides Java Enterprise Edition web services. For these Application Notes, a sample Avaya CPM web client application available in the CPM SDK was deployed to the BEA Weblogic Application Server via BEA Workshop. This client application was used to verify the Avaya CPM Simple Object Access Protocol (SOAP) web services interaction between the Avaya CPM and BEA Weblogic Application Server.

# 1. Introduction

These Application Notes describe the steps required to package and deploy Avaya Communications Process Manager (CPM) sample SDK web application on a BEA Weblogic Application Server using BEA Workshop and the BEA Weblogic administrative console.

BEA Weblogic is an application server for use with Java 2 Platform, Enterprise Edition (J2EE) and Enterprise Java Beans (EJB). For these Application Notes, a sample Avaya CPM web client application available in the Avaya CPM SDK was recompiled, and the resulting Web Application Archive (WAR) file was deployed to the BEA Weblogic Application Server. This web client application was used to verify the CPM Simple Object Access Protocol (SOAP) Web services interaction between the Avaya CPM and the BEA Weblogic Application Server platform. The CPM sample web client application project was written in Java code and is platform independent. These Application Notes illustrate how to recompile and deploy this sample CPM web client application to the BEA Weblogic Application Server using BEA Workshop and the BEA Weblogic administrative console.

The Avaya CPM SDK sample web client application supports the following Avaya CPM web services:

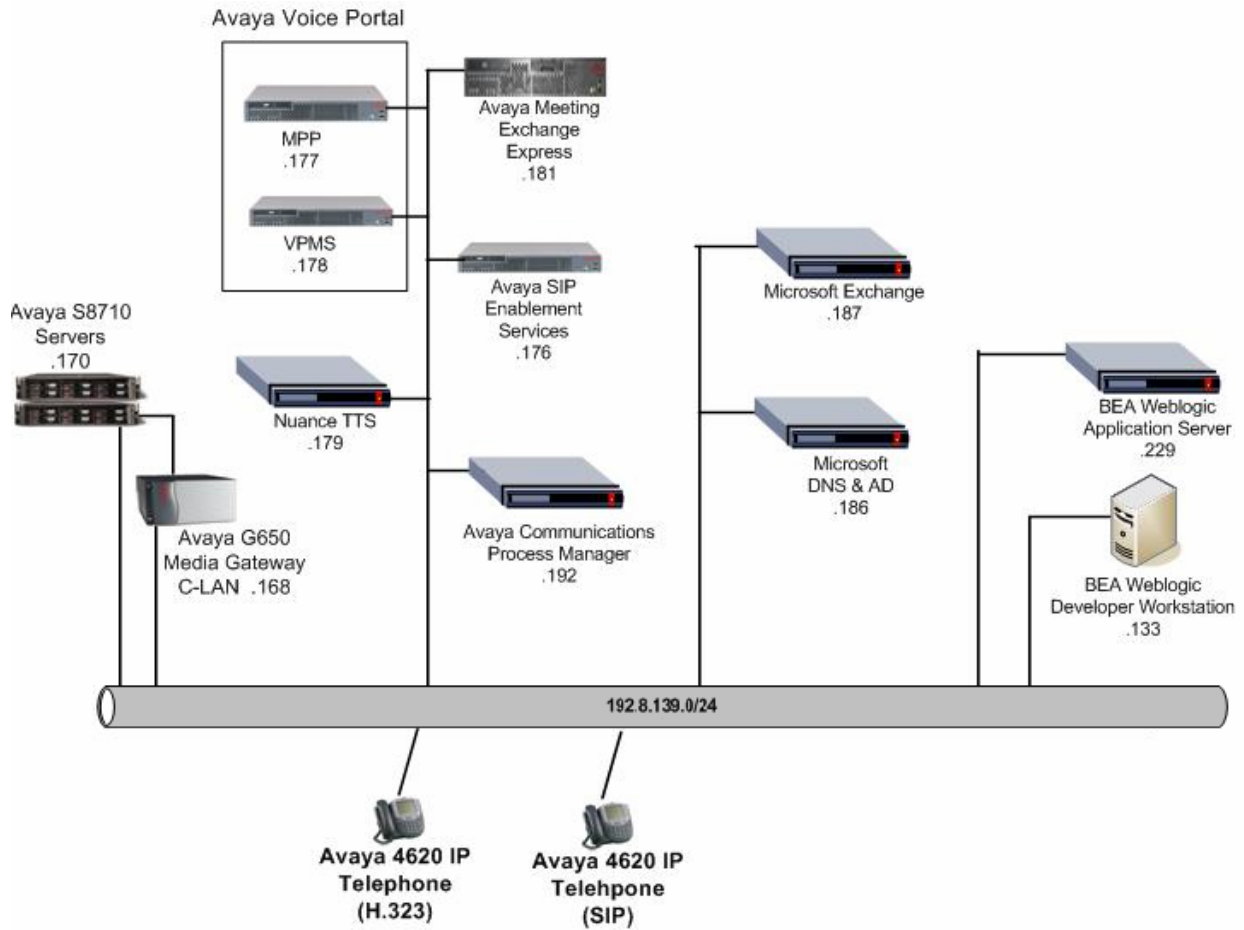
- **Advisory:** This web service initiates an outbound advisory request to a list of recipients for them to acknowledge receipt of the notification.
- **Notify and Respond:** This web service initiates an outbound notification with a set of questions to a list of recipients and waits for them to respond to the notification.
- **Notify and Conference:** This web service initiates an outbound notification to a list of recipients. When a notified user answers the phone, the service provides contextual information about the exception conference and asks if the caller wants to join the conference.
- **Find and Call:** This web client application uses a recipient list to create either a two-party call or an on demand conference.

For detailed information on the sample Avaya SDK web client application refer to [8] in the Additional References section.

## 2. Configuration

**Figure 1** provides an overview of the network used in the sample configuration. This sample configuration consists of Avaya S8710 Servers with an Avaya G650 Media Gateway running Avaya Communication Manager, Avaya SIP Enablement Services (SES), Avaya Voice Portal, Nuance Text to Speech server, Avaya Meeting Exchange Express, Avaya Communications Process Manager and the BEA Weblogic Application Server. The Avaya 4600 Series IP Telephones (both H.323 and SIP) are registered with Avaya Communication Manager (running on the Avaya S8710 Server) and Avaya SIP Enablement Services (SES), respectively. Avaya

SIP telephones were configured as Off-PBX stations (OPS). Avaya Voice Portal serves as an interactive voice response system for converting CPM text message to voice.



**Figure 1: Network Configuration**

### 3. Equipment and Software Validated

The following equipment and software were used for the sample configuration described in these Application Notes:

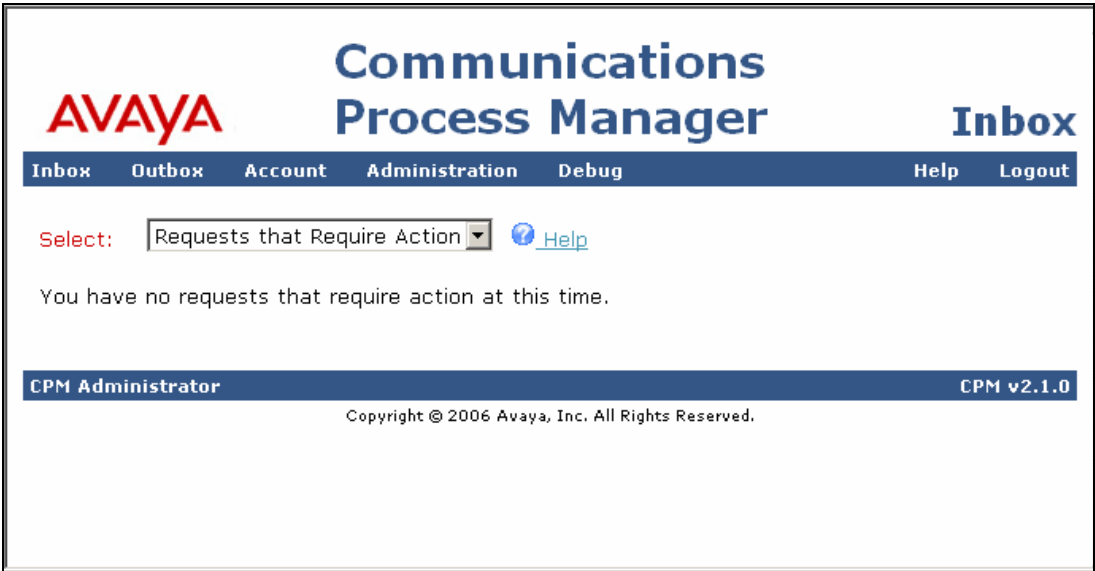
Equipment & Software	Version
Avaya S8710 Server	Avaya Communication Manager 4.0.1 (R014x.00.1.731.2)
Avaya G650 Media Gateway	
Avaya 4600 Series IP Telephones	2.2.2 (SIP)
Avaya 4600 Series IP Telephones	2.8 (H.323)
Avaya Communication Process Manager	R2.1.53
Avaya SIP Enablement Services	4.0-04.0.033.6
Avaya Voice Portal VPMS MPP	4.0.0.0.2901
Avaya Meeting Exchange Express	2.5.60.0
Nuance TTS server Real speak	4.0.10
Avaya Communications Process Manager SDK	2.1.53
BEA Weblogic Application Server <ul style="list-style-type: none"> <li>Microsoft Windows 2003 Server</li> <li>BEA JRockit jvm</li> </ul>	Version 10.0 (Build id: 933139) Service Pack 2 build R26.4.0-63_CR302700- 72606-1.5.0_06-20061127-1108- win-ia32
BEA Workshop for Weblogic	Version 10.0 (Build id.: 933139)
Microsoft Active Directory and DNS Server <ul style="list-style-type: none"> <li>Microsoft Windows 2003 Server</li> </ul>	Service Pack 2

**Table 1: Equipment /Software List**

## 4. Configure Avaya Communications Process Manager

These Application Notes assume that the Avaya CPM software and the license file have already been previously installed. It is further assumed that Avaya Communication Manager, Avaya SES, Avaya Voice Portal, and Avaya Meeting Exchange Express (as shown in Fig. 1) have already been configured and are operational with Avaya CPM. For additional information on these installation tasks, refer to [1], [2], [3], [4], and [5] in Section 11.

This section describes the steps that are required for adding users to the Avaya Communications Process Manager for use with the sample SDK web application described in later sections.

Step	Description
1.	On the Avaya CPM server, launch a web browser and enter the URL <b>http://&lt; CPM server Name or IP address &gt;/VIA</b> . When prompted for a user name and password, enter the credentials of the server administrator account.
2.	<p>The Communications Process Manager Administration web portal page opens. To add a new user, click Account.</p> 

3. The **Account Home** page opens. To create a new user account, click **Create Account**.

The screenshot shows the Avaya Communications Process Manager Account Home page. At the top left is the AVAYA logo. To its right is the title 'Communications Process Manager' in large blue font, and further right is the word 'Account' in blue. Below these is a dark blue navigation bar with white text for 'Inbox', 'Outbox', 'Account', 'Administration', 'Debug', 'Help', and 'Logout'. The main content area has a heading 'Account Home' with a blue 'Help' link. Below this, there are several sections: 'Account Home' with a sub-heading 'The services on the left allow you to:' followed by a bulleted list: 'Specify personal information including your time zone', 'Change your PIN', and 'Specify your preferences for being contacted'. Other sections include 'Account Information', 'Change PIN', 'Points Of Contact', 'Time Profiles', 'Notification Profiles', 'Notification Filters', 'List Management', and 'Create Account'. A 'User Search' section is also present, with a sub-heading 'User Search' and a paragraph explaining the search functionality, including a search input field.

4.	<p>The <b>Create Account</b> page opens. The <b>Create Account</b> screen is shown in <b>Steps 5 and 6</b>. Enter the following values to add a new user to the CPM LDAP directory.</p> <ul style="list-style-type: none"><li>• <b>Administrator:</b> To give the user administrator access, select <b>Yes</b>.</li><li>• <b>CPM User:</b> To make this user a licensed Communications Process Manager user, select <b>Yes</b>. Licensed Communications Process Manager users can log in to the Communications Process Manager portal and receive notifications.</li><li>• <b>Handle:</b> Enter the user's E-mail handle (e.g., <b>billsmith</b>).</li><li>• <b>ID Number:</b> A unique ID number (e.g., <b>39101</b>) for the user. Can be an employee number or something similar. This ID number serves as the user's account number and initial PIN.</li><li>• <b>Last Name:</b> Enter User's last name (e.g., <b>Smith</b>).</li><li>• <b>Common Name(s):</b> Communications Process Manager typically uses the first value of the common name in interactions with the user.</li><li>• <b>Phone Number:</b> Telephone number at which Communications Process Manager contacts the user (e.g., <b>sip:39101@cebp-avaya.com</b>.) This is the default telephone number at which Communications Process Manager contacts the user.</li><li>• <b>Advisory Service:</b> Select <b>Yes</b>.</li><li>• <b>Click To Find Service:</b> Select <b>Yes</b>.</li><li>• <b>Notification And Response Service:</b> Select <b>Yes</b>.</li><li>• <b>Notify And Conference Service:</b> Select <b>Yes</b>.</li></ul> <p>Click <b>Save</b>.</p>
----	--

5.

Roles	
<b>Administrator:</b>	<input checked="" type="radio"/> Yes <input type="radio"/> No
<b>CPM User:</b>	<input checked="" type="radio"/> Yes <input type="radio"/> No
Attributes	
<b>*Handle:</b>	<input type="text" value="Bill"/>
<b>*ID Number:</b>	<input type="text" value="39101"/>
<b>Display Name:</b>	<input type="text"/>
<b>First Name:</b>	<input type="text"/>
<b>*Last Name:</b>	<input type="text" value="Smith"/>
<b>*Common Name(s):</b>	<input type="text" value="Bill"/>
<b>Phone Number:</b>	<input type="text" value="sip:39101@cebp-avaya.com"/>
<b>Mobile Phone Number:</b>	<input type="text"/>
<b>Fax Number:</b>	<input type="text"/>
<b>Pager Number:</b>	<input type="text"/>
<b>Electronic Mail Address:</b>	<input type="text"/>
<b>Honorific:</b>	<input type="text" value="Mr."/> ▼
<b>Title:</b>	<input type="text"/>
<b>Affiliation:</b>	<input type="checkbox"/> Employee ▼
<b>Manager:</b>	<input type="text"/>

6.	<b>Department:</b>	<input type="text"/>
	<b>Organization:</b>	<input type="text"/>
	<b>Room:</b>	<input type="text"/>
	<b>Street:</b>	<input type="text"/>
	<b>City:</b>	<input type="text"/>
	<b>State:</b>	<input type="text"/>
	<b>Postal Code:</b>	<input type="text"/>
	<b>Language:</b>	US English ▾
	<b>Time Zone:</b>	<input checked="" type="radio"/> America/New_York ▾
		<input type="radio"/> GMT <input type="text" value="00"/> : <input type="text" value="00"/>
	<b>Applications</b>	
	<b>Advisory Service:</b>	<input checked="" type="radio"/> Yes <input type="radio"/> No
	<b>Click To Find Service:</b>	<input checked="" type="radio"/> Yes <input type="radio"/> No
	<b>Notification And Response Service:</b>	<input checked="" type="radio"/> Yes <input type="radio"/> No
<b>Notify And Conference Service:</b>	<input checked="" type="radio"/> Yes <input type="radio"/> No	
<input type="button" value="Clear"/>		
<input type="button" value="Save"/>		

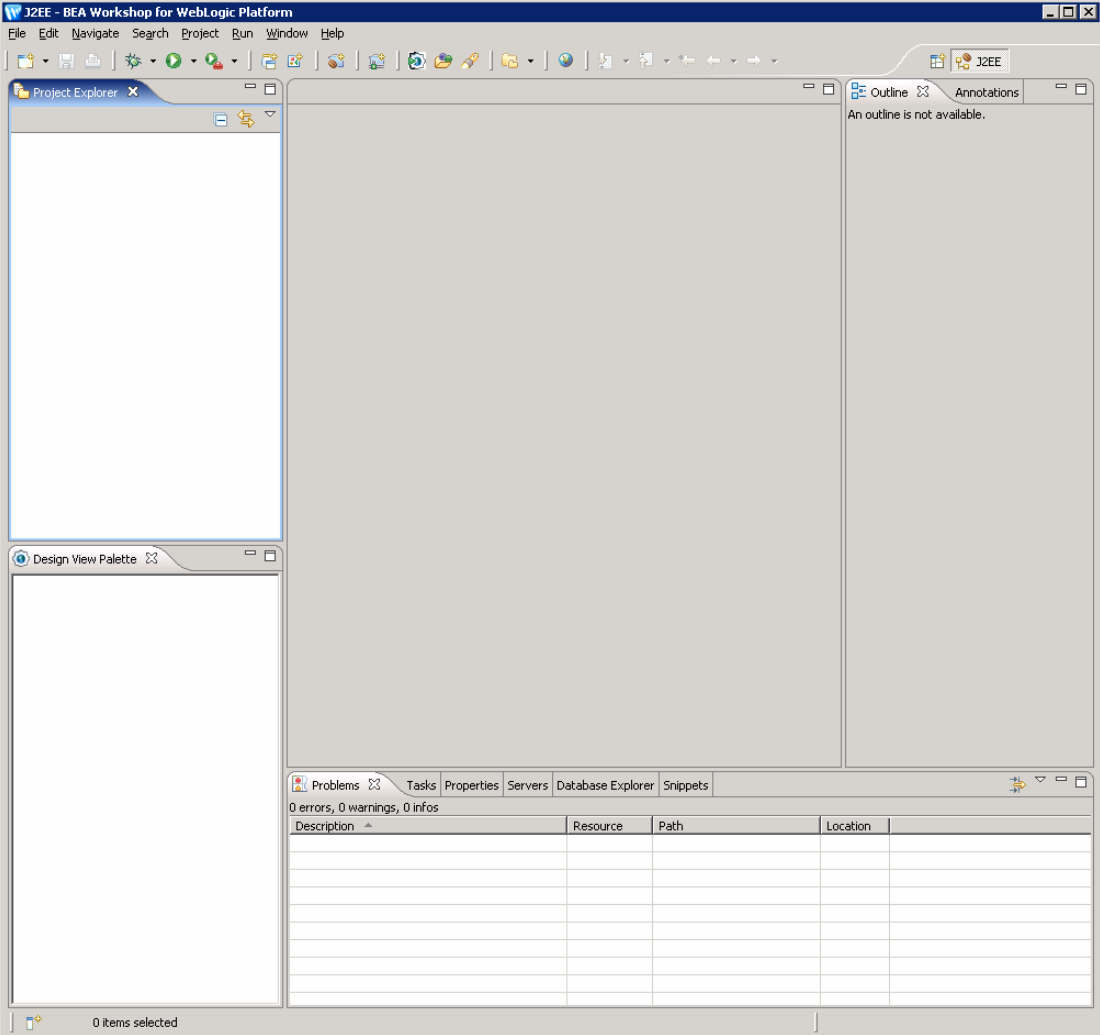
## 5. Compile Sample Avaya Communications Process Manager Web Client Application

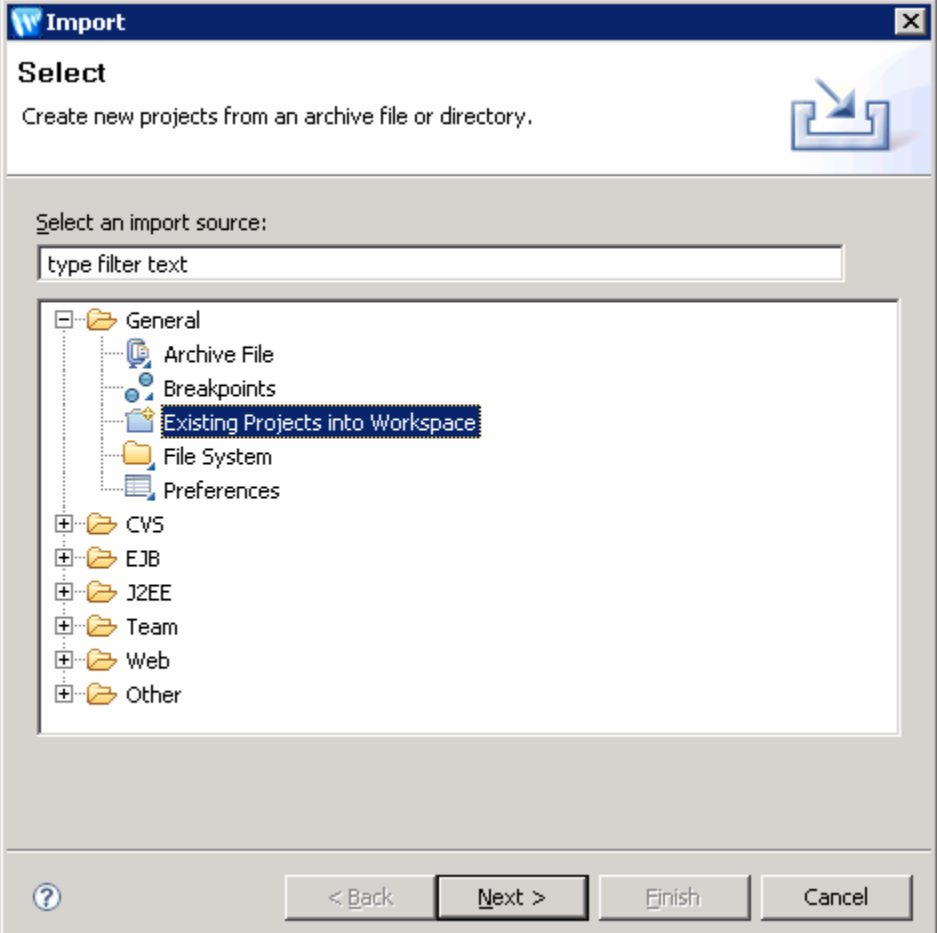
This section presents the steps to import the Avaya CPM SDK web client application, and compile the web application using BEA Workshop.

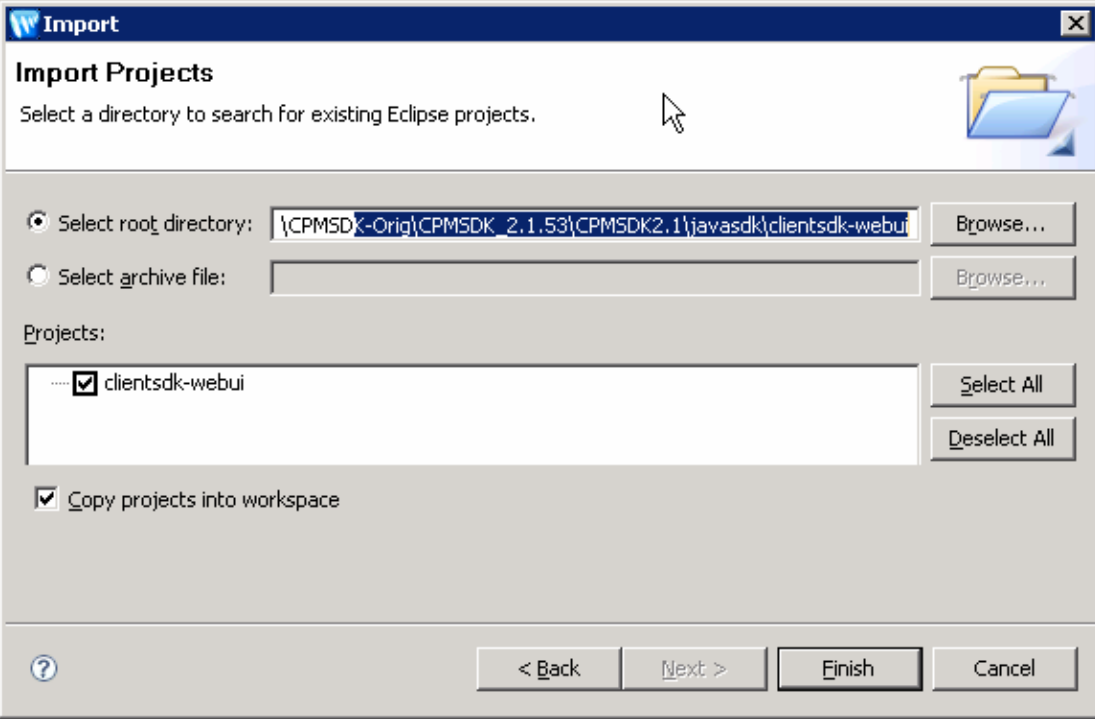
These Application Notes assume that BEA Weblogic application server is installed and operational on a server. In addition, it is assumed that BEA Workshop is installed on a user workstation. The BEA Workshop installation also installs BEA Weblogic as part of the installation. For additional information on BEA Weblogic and BEA Workshop, refer to [9] – [14] in the Additional References section.

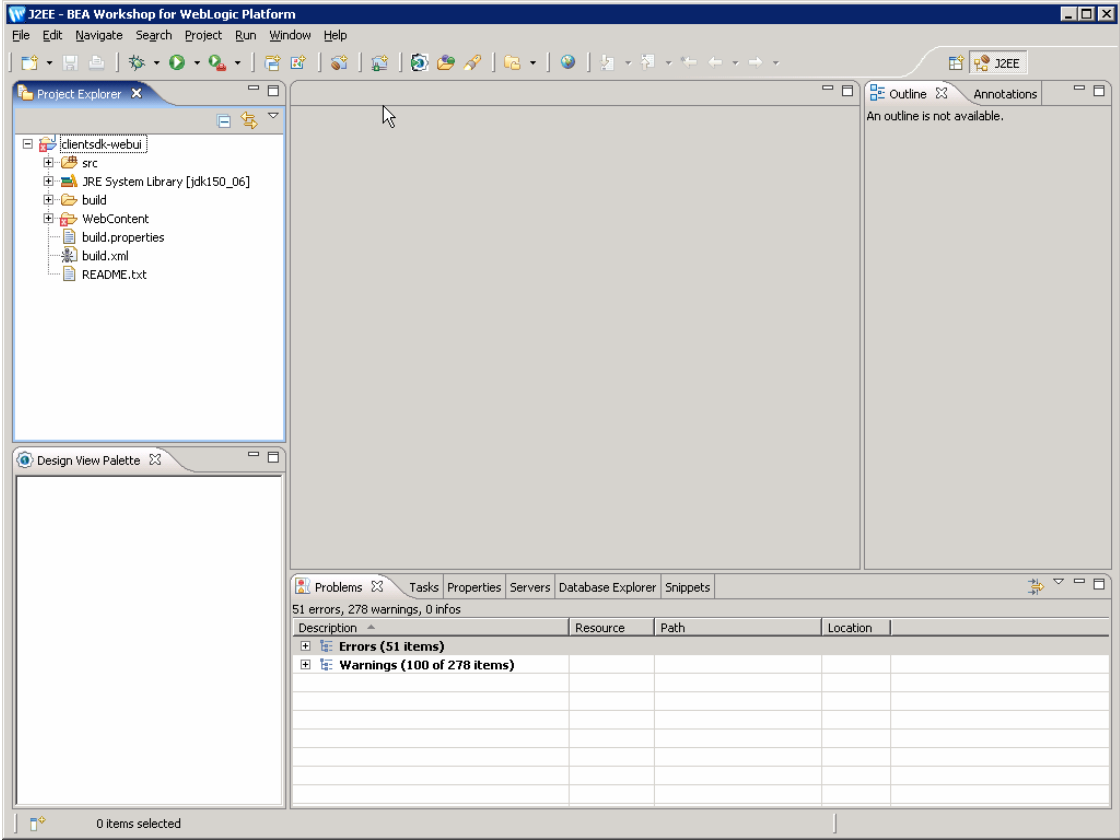
The CPM Client SDK is available for the registered Devconnect members. It can be downloaded from the following URL: <http://www.avaya.com/devconnect>. For more information on the sample Avaya SDK web client application, refer to [7] in the Additional References section.

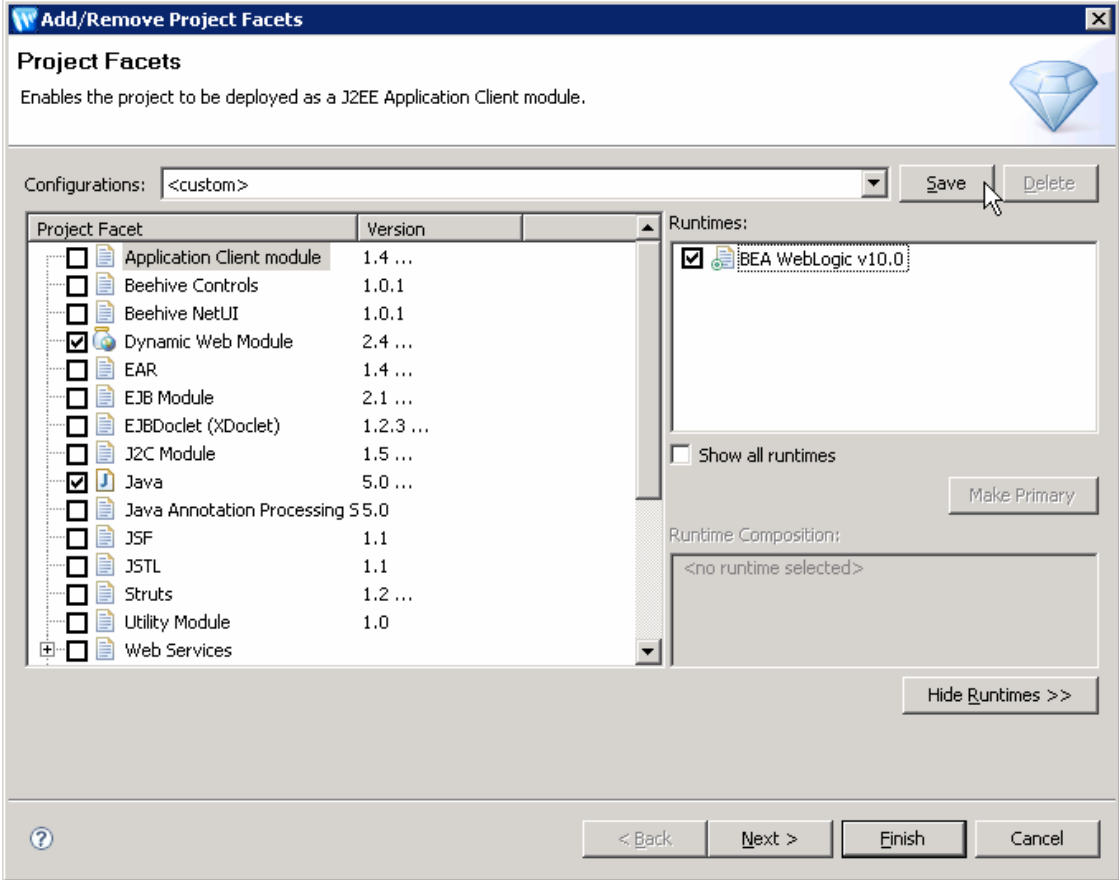
Step	Description
1.	Copy the CPM SDK zip file onto the workstation where the BEA Workshop is installed. Expand the zip file, and note the location of the expanded directory.

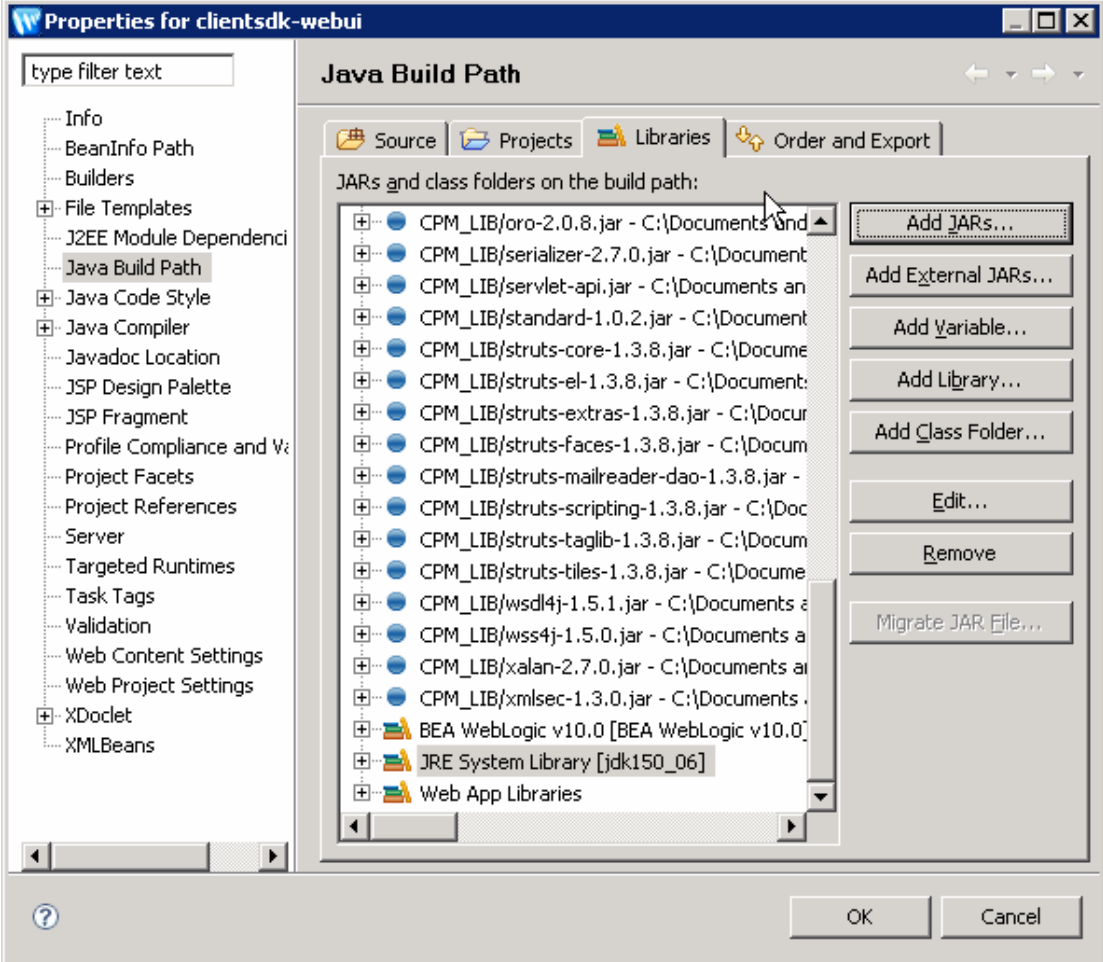
Step	Description
2.	<p>Launch BEA Workshop for Weblogic. On the BEA Weblogic server, select <b>Start → All Programs → BEA Products → Workshop for Weblogic Platform</b>. The <b>Workspace Launcher</b> dialog box opens (not shown). Select an existing, or create a new, workspace for the CPM Client Application. Click <b>OK</b> to open the <b>J2EE – BEA Workshop for Weblogic Platform</b> (referred to in the remainder of this document as <b>BEA Workshop</b>) window. If you created a new project the <b>Welcome</b> tab will be displayed (not shown). Close this tab to display the <b>J2EE</b> perspective as shown below.</p> 

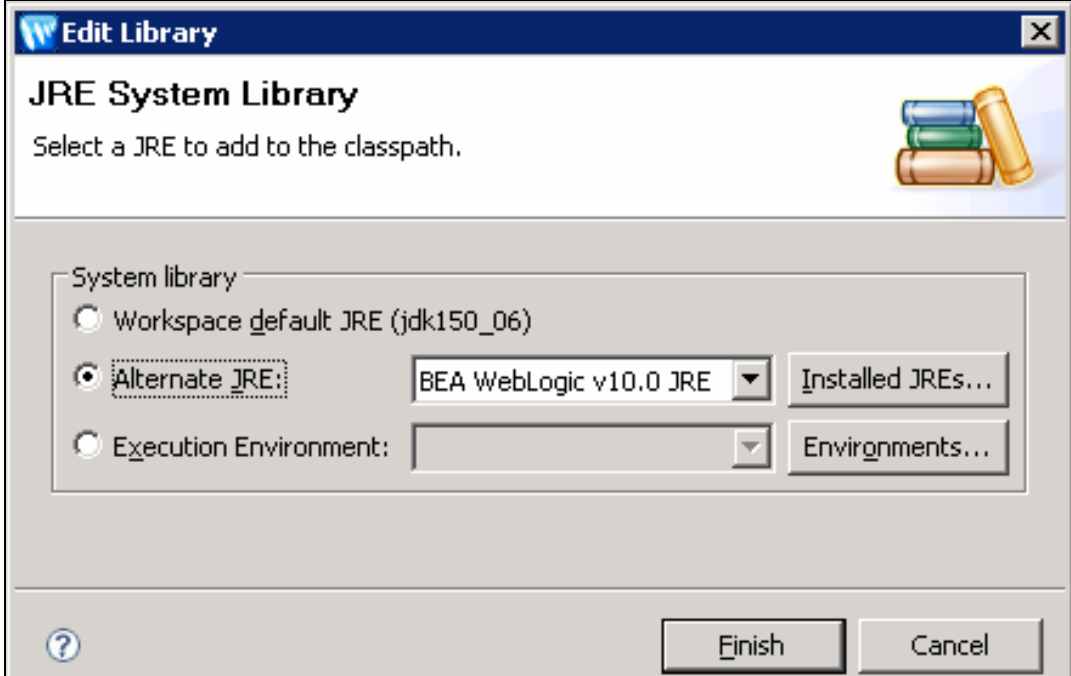
Step	Description
3.	<p>Import the <b>clientsdk-webui</b> project in BEA Workshop. Select the <b>File → Import...</b> menu item to open the <b>Select</b> window. Expand the <b>General</b> folder, and select the <b>Existing Projects into Workspace</b> item, as shown below. Click <b>Next</b>.</p> 

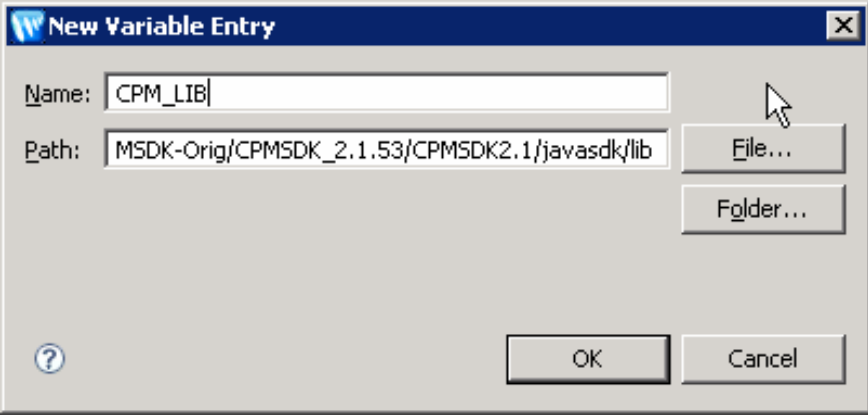
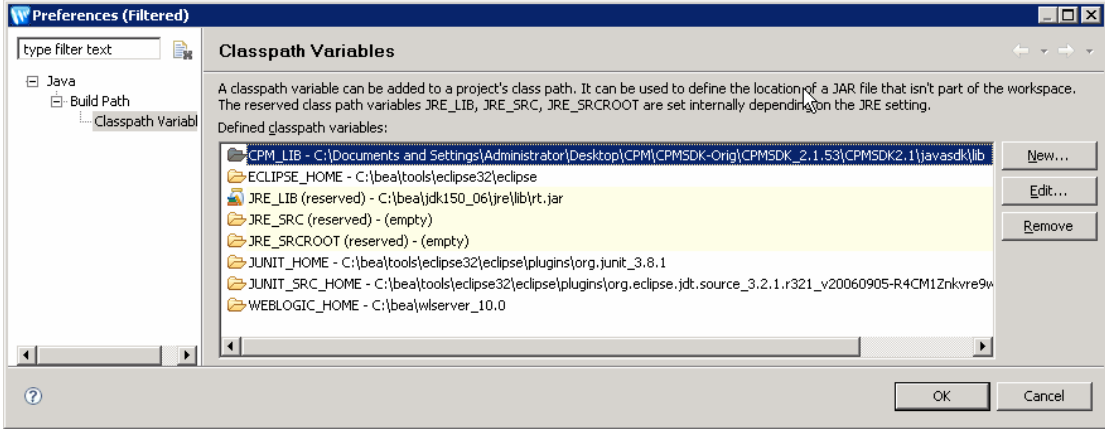
Step	Description
4.	<p>The <b>Import Project</b> window opens as shown below. Select the <b>Select root directory:</b> radio button. Then click <b>Browse...</b> to open the <b>Browse For Folder</b> window (not shown). Navigate to and select the <b>clientsdk-webui</b> directory in the CPM SDK directory expanded in <b>Step 1</b> above. Click <b>OK</b>. Verify the <b>clientsdk-webui</b> checkbox is checked in the <b>Projects:</b> area. Check the <b>Copy projects into workspace</b> checkbox at the bottom of the window. Click <b>Finish</b> to import the project, and to return to the <b>BEA Workshop</b> window.</p> 

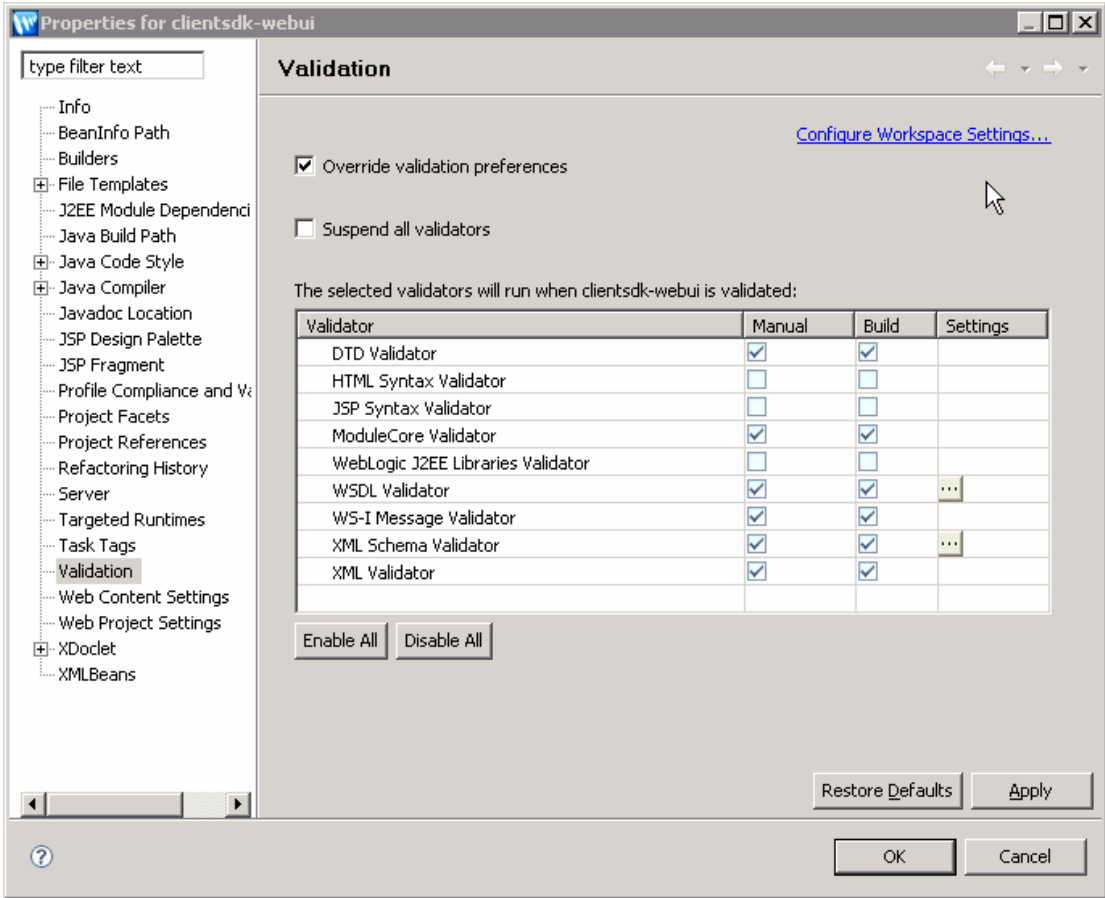
Step	Description
<p>5.</p>	<p>Verify the <b>clientsdk-webui</b> project now appears in the <b>Project Explorer</b> tab of the <b>BEA Workshop</b> window, as shown below.</p> <p><b>BEA Workshop</b> is configured to automatically build the project. Therefore, if there are issues with building the project, they will be listed in the <b>Problems</b> tab of the <b>BEA Workshop</b> window. Select the <b>Problems</b> tab near the bottom of the <b>BEA Workshop</b> window. Note there are a number of errors and warnings listed in this tab, as shown below. The following steps will resolve these issues.</p>  <p>The screenshot shows the BEA Workshop interface. The Project Explorer on the left displays the project structure for 'clientsdk-webui', including folders for 'src', 'build', and 'WebContent', and files like 'build.properties', 'build.xml', and 'README.txt'. The Problems tab at the bottom shows a summary of '51 errors, 278 warnings, 0 infos' and a table with columns for Description, Resource, Path, and Location. The table lists 'Errors (51 items)' and 'Warnings (100 of 278 items)'.</p>
<p>6.</p>	<p>Right click on the <b>clientsdk-webui</b> project in the Project Explorer tab. In the drop down menu that opens select the <b>Properties</b> menu item (not shown). The <b>Properties for clientsdk-webui</b> window opens (Not shown).</p>
<p>7.</p>	<p>In the left frame of the Properties for clientsdk-webui window select the <b>Project Facets</b> item. The <b>Project Facets</b> frame will open on the right side of the window, (not shown). Click the <b>Add/Remove Project Facets ...</b> button.</p>

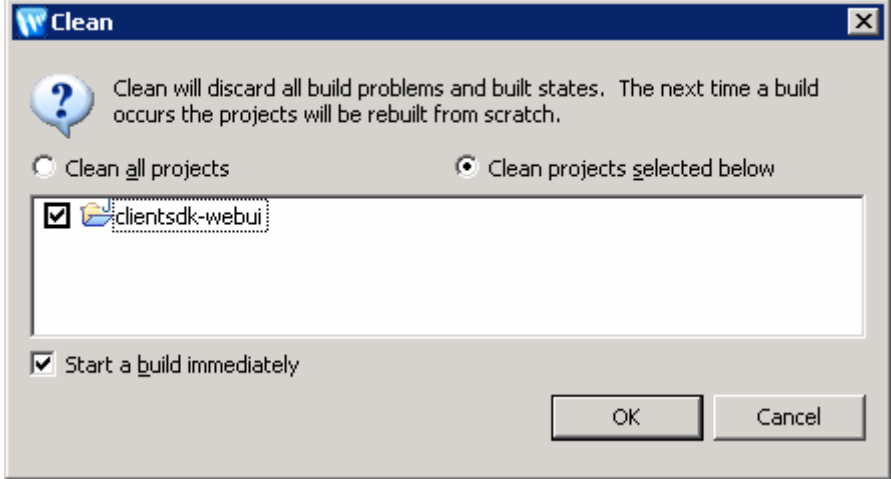
Step	Description
8.	<p>The <b>Add/Remove Project Facets</b> window opens (not shown). Click the &lt;&lt;<b>Show Runtimes</b> button to open the <b>Runtimes:</b> and <b>Runtime Composition:</b> (grayed out) frames in the <b>Add/Remove Project Facets</b> window. Check the BEA Weblogic v10.0 checkbox in the <b>Runtimes:</b> frame.</p> <p>Check the <b>Java</b> and the <b>Dynamic Web Module</b> check boxes in the <b>Project Facet</b> frame of the window. Click <b>Next &gt;</b> to open the <b>web module</b> window (not shown). Leave all values as is, and click <b>Finish</b> to return to the <b>Properties for clientsdk-webui</b> window (not shown). The projects facets selected above will be listed in the right frame of the <b>Properties for clientsdk-webui</b> window (not shown).</p> 

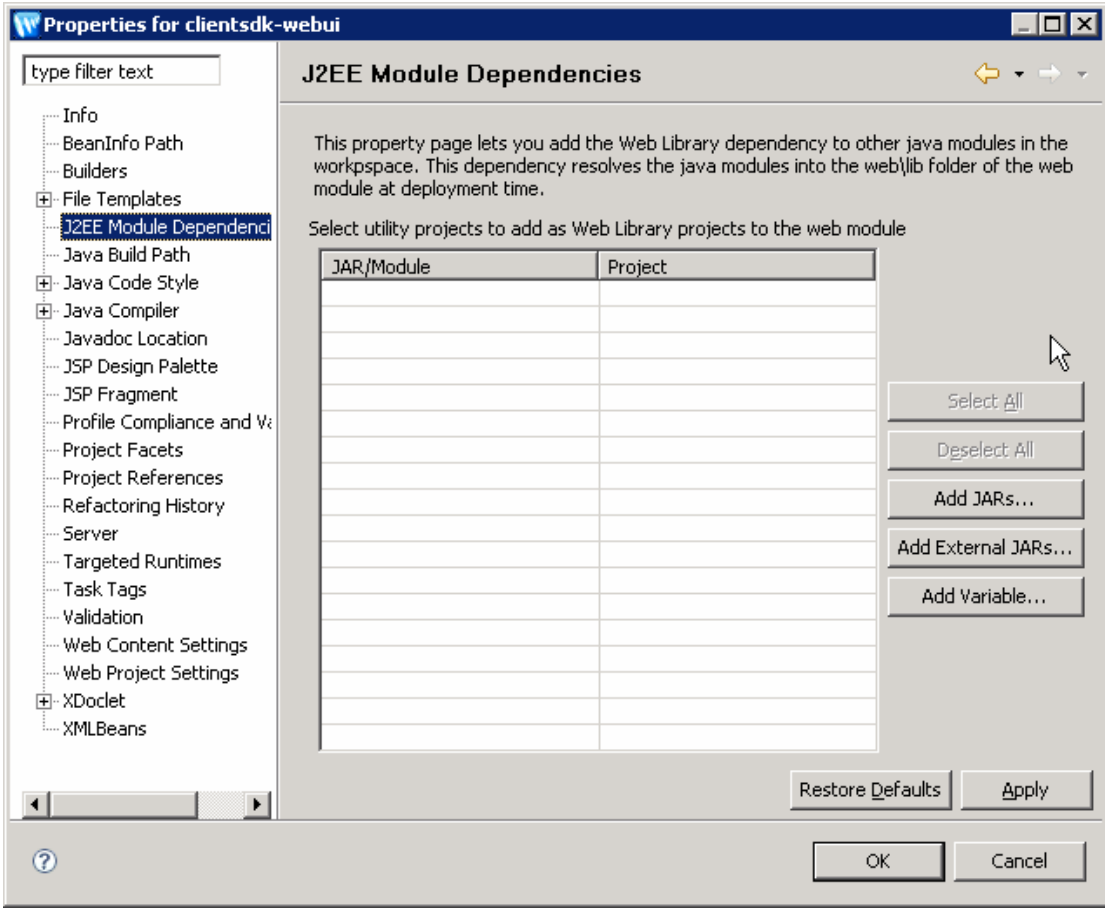
Step	Description
9.	<p>In the left frame of the <b>Properties for clientsdk-webui</b> window select the <b>Java Build Path</b> item. The <b>Java Build Path</b> frame will open in the right frame, shown below. In the <b>Java Build Path</b> frame, select the <b>Libraries</b> tab. In the <b>Libraries</b> tab scroll down and select the <b>JRE System Library</b> entry. If the jdk is <b>jdk150_06</b>, select the entry and click the <b>Edit...</b> button.</p>  <p>The screenshot shows the 'Properties for clientsdk-webui' dialog box. On the left is a tree view with 'Java Build Path' selected. The main area is titled 'Java Build Path' and has four tabs: 'Source', 'Projects', 'Libraries', and 'Order and Export'. The 'Libraries' tab is active, displaying a list of 'JARs and class folders on the build path'. The list includes various CPM_LIB JARs and the 'JRE System Library [jdk150_06]'. The 'Edit...' button is highlighted, indicating the next step in the procedure.</p>

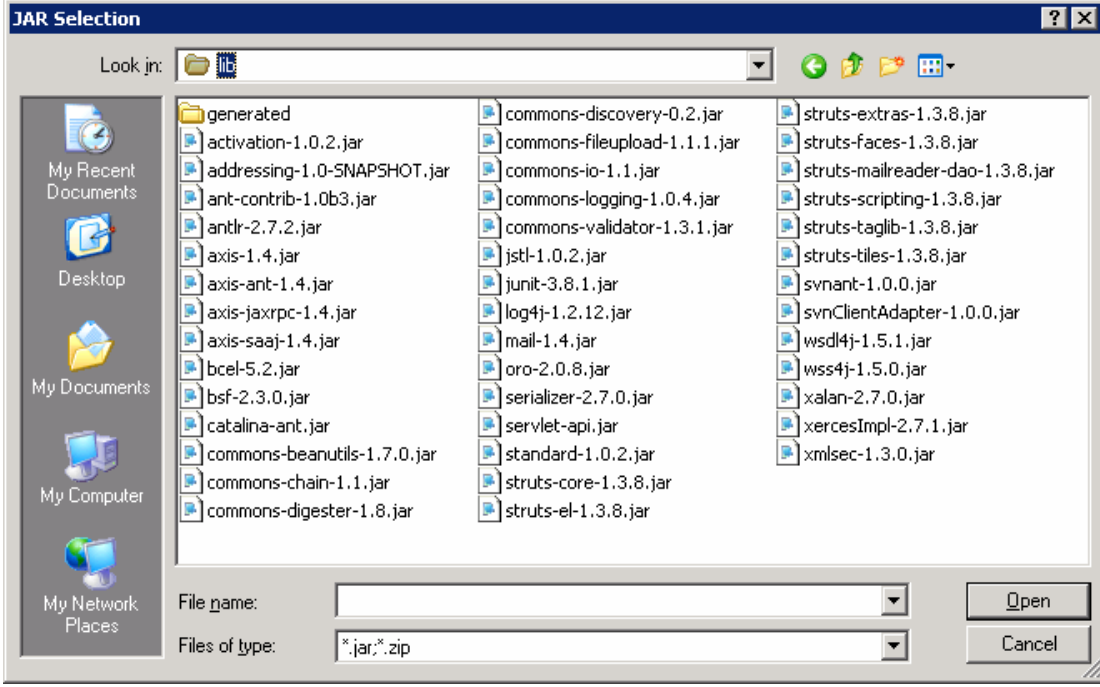
Step	Description
10.	<p>The <b>JRE System Library</b> window opens. Select the <b>Alternate JRE:</b> radio button. Select the <b>BEA Weblogic v10.0 JRE</b> from the drop down list. Click the <b>Finish</b> button to select the BEA jrocket jre.</p>  <p>The screenshot shows a dialog box titled "Edit Library" with a close button in the top right. Below the title bar, it says "JRE System Library" and "Select a JRE to add to the classpath." There is an icon of three stacked cylinders (blue, green, orange) on the right. Under the heading "System library", there are three radio buttons: "Workspace default JRE (jdk150_06)", "Alternate JRE:", and "Execution Environment:". The "Alternate JRE:" radio button is selected. To its right is a dropdown menu showing "BEA WebLogic v10.0 JRE" and a button labeled "Installed JREs...". Below the "Execution Environment:" radio button is another dropdown menu and a button labeled "Environments...". At the bottom of the dialog, there is a help icon (a question mark in a circle) on the left, and "Finish" and "Cancel" buttons on the right.</p>

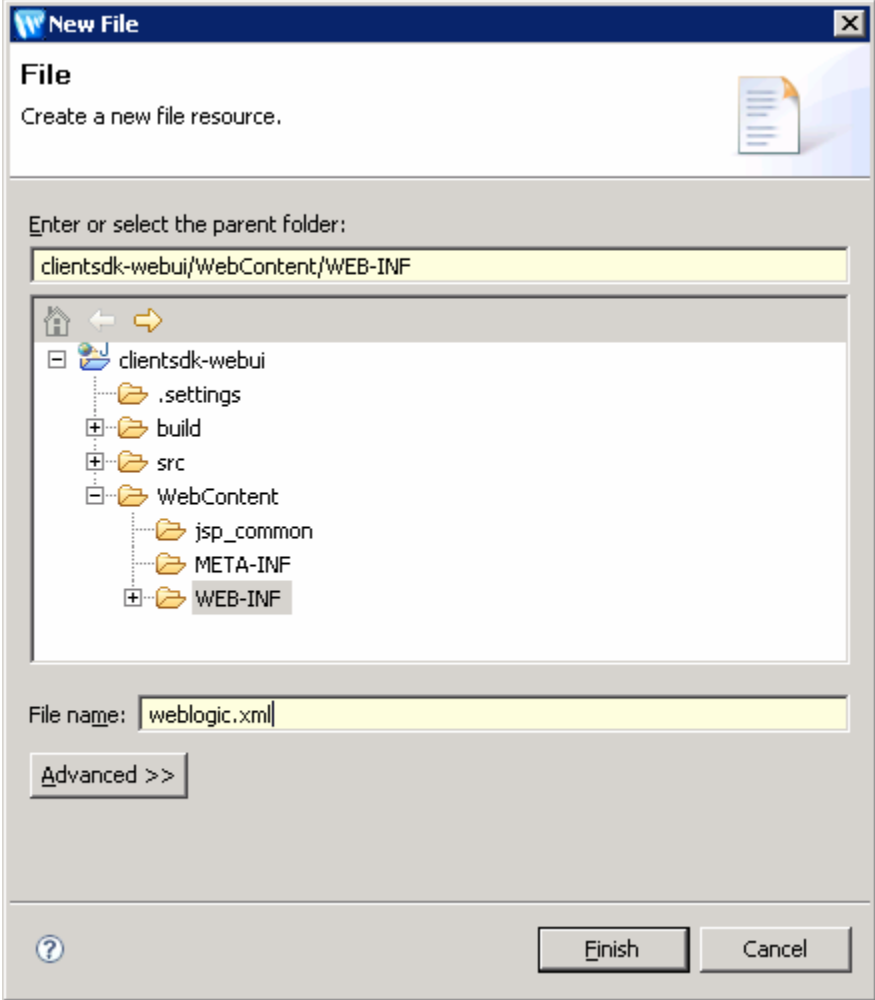
Step	Description
11.	<p>Click the <b>Add Variable...</b> button, in the <b>Properties Java Built Path</b> frame (shown in <b>Step 9</b> above). The <b>New Variable Classpath Entry</b> window opens (not shown). Click the <b>Configure Variables...</b> button to open the <b>Preferences (Filtered)</b> window (not shown). Click the <b>New...</b> button to open the <b>New Variable Entry</b> window. In the <b>Name:</b> text box enter <b>CPM_LIB</b>. Click on the <b>Folder...</b> button to open the <b>Folder Selection</b> window (Not shown). Navigate to and select the <b>lib</b> folder in the CPM SDK directory expanded in <b>step 1</b> above. Click <b>OK</b> to return to the <b>New Variable Entry</b> window. Click <b>OK</b> again to add the new variable to the java build path.</p> 
12.	<p>The <b>Preferences (Filtered)</b> window now shows the <b>CPM_LIB</b> variable listed in the <b>Defined classpath variables:</b> list. Click <b>OK</b> to update the <b>clientsdk-webui</b> projects classpath. A dialog box opens warning that the classpath variable has changed and a full rebuild is necessary for the changes to take affect (not shown). Click <b>Yes</b> to do a full rebuild the project.</p> 
13.	<p>After the rebuild, the <b>New Variable Entry</b> window remains open, click <b>Cancel</b> to close this window. Click <b>OK</b> to close the <b>Properties for clientsdk-webui</b> window. The <b>Problems</b> tab of the <b>BEA Workshop</b> window will show one error and a number of warnings that still need to be resolved.</p>

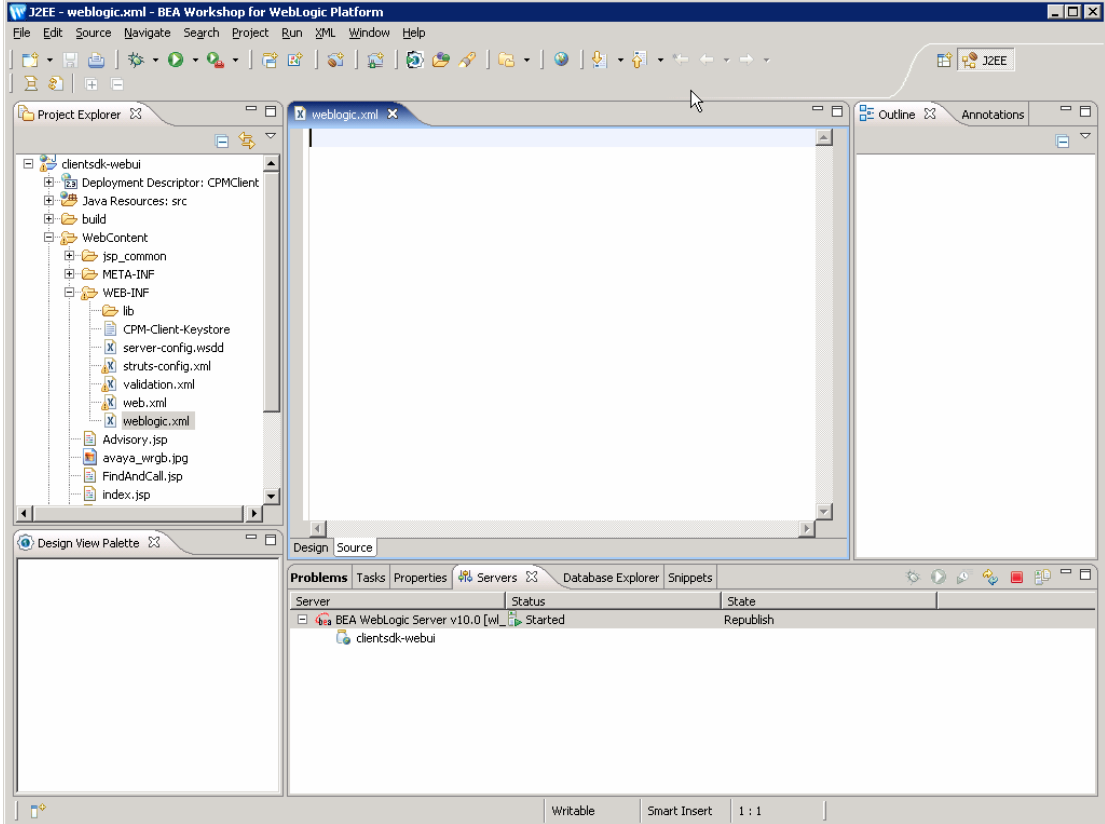
Step	Description																																								
14.	<p>Right click on the <b>clientsdk-webui</b> project in the Project Explorer tab. In the drop down menu that opens select the <b>Properties</b> menu item (not shown). The <b>Properties for clientsdk-webui</b> window opens.</p> <p>In the left frame of the <b>Properties for clientsdk-webui</b> window select the <b>Validation</b> item. The <b>Validation</b> frame will open in the right frame. Check the <b>Override validation preferences</b> checkbox. Uncheck both the <b>Manual</b> and <b>Build</b> checkboxes for both the <b>HTML Syntax Validator</b> and the <b>JSP Syntax Validator</b>. Then click <b>OK</b> to return to the <b>BEA Workshop</b> window.</p>  <table border="1" data-bbox="609 892 1360 1192"> <thead> <tr> <th>Validator</th> <th>Manual</th> <th>Build</th> <th>Settings</th> </tr> </thead> <tbody> <tr> <td>DTD Validator</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> <tr> <td>HTML Syntax Validator</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td></td> </tr> <tr> <td>JSP Syntax Validator</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td></td> </tr> <tr> <td>ModuleCore Validator</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> <tr> <td>WebLogic J2EE Libraries Validator</td> <td><input type="checkbox"/></td> <td><input type="checkbox"/></td> <td></td> </tr> <tr> <td>WSDL Validator</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>...</td> </tr> <tr> <td>WS-I Message Validator</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> <tr> <td>XML Schema Validator</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td>...</td> </tr> <tr> <td>XML Validator</td> <td><input checked="" type="checkbox"/></td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> </tbody> </table>	Validator	Manual	Build	Settings	DTD Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		HTML Syntax Validator	<input type="checkbox"/>	<input type="checkbox"/>		JSP Syntax Validator	<input type="checkbox"/>	<input type="checkbox"/>		ModuleCore Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		WebLogic J2EE Libraries Validator	<input type="checkbox"/>	<input type="checkbox"/>		WSDL Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	...	WS-I Message Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		XML Schema Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	...	XML Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Validator	Manual	Build	Settings																																						
DTD Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																							
HTML Syntax Validator	<input type="checkbox"/>	<input type="checkbox"/>																																							
JSP Syntax Validator	<input type="checkbox"/>	<input type="checkbox"/>																																							
ModuleCore Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																							
WebLogic J2EE Libraries Validator	<input type="checkbox"/>	<input type="checkbox"/>																																							
WSDL Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	...																																						
WS-I Message Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																							
XML Schema Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	...																																						
XML Validator	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>																																							

Step	Description
15.	<p>Force the project to recompile by selecting the <b>Project → Clean...</b> menu item. The <b>Clean</b> window opens. Select either radio button. If the <b>Clean projects selected below</b> radio button, then check the <b>clientsdk-webui</b> project in the list. Click the <b>OK</b> button to clean and force the project to rebuild. BEA Workshop is set by default to automatically build projects, therefore when clean is selected, a rebuild also occurs.</p> 

Step	Description
16.	<p>After completing the above steps the web application build will succeed. However, the deployment of the web application will fail due to dependencies on java modules that have not been resolved.</p> <p>To resolve the java module dependencies, right click on the <b>clientsdk-webui</b> project in the <b>Project Explorer</b> of <b>BEA Workshop</b> to open the <b>Properties for clientsdk-webui</b> window. In the left frame of the <b>Properties for clientsdk-webui</b> window select the <b>J2EE Module Dependencies</b> item. In the <b>J2EE Module Dependencies</b> frame on the right, click the <b>Add External JARs...</b> button.</p> 

Step	Description
17.	<p>The <b>JAR Selection</b> window opens. Navigate to the <b>lib</b> folder in the CPM SDK directory expanded in <b>step 1</b> above. Select all the <b>jar</b> files in this directory. Click <b>Open</b> to add the J2EE module dependencies. The addition of the J2EE module dependencies may take 10-15 minutes or more to complete.</p>  <p>The screenshot shows a 'JAR Selection' dialog box with a 'Look in:' field set to 'lib'. The file list contains the following items:</p> <ul style="list-style-type: none"> <li>generated</li> <li>activation-1.0.2.jar</li> <li>addressing-1.0-SNAPSHOT.jar</li> <li>ant-contrib-1.0b3.jar</li> <li>antlr-2.7.2.jar</li> <li>axis-1.4.jar</li> <li>axis-ant-1.4.jar</li> <li>axis-jaxrpc-1.4.jar</li> <li>axis-saaj-1.4.jar</li> <li>bcel-5.2.jar</li> <li>bsf-2.3.0.jar</li> <li>catalina-ant.jar</li> <li>commons-beanutils-1.7.0.jar</li> <li>commons-chain-1.1.jar</li> <li>commons-digester-1.8.jar</li> <li>commons-discovery-0.2.jar</li> <li>commons-fileupload-1.1.1.jar</li> <li>commons-io-1.1.jar</li> <li>commons-logging-1.0.4.jar</li> <li>commons-validator-1.3.1.jar</li> <li>jstl-1.0.2.jar</li> <li>junit-3.8.1.jar</li> <li>log4j-1.2.12.jar</li> <li>mail-1.4.jar</li> <li>oro-2.0.8.jar</li> <li>serializer-2.7.0.jar</li> <li>servlet-api.jar</li> <li>standard-1.0.2.jar</li> <li>struts-core-1.3.8.jar</li> <li>struts-el-1.3.8.jar</li> <li>struts-extras-1.3.8.jar</li> <li>struts-faces-1.3.8.jar</li> <li>struts-mailreader-dao-1.3.8.jar</li> <li>struts-scripting-1.3.8.jar</li> <li>struts-taglib-1.3.8.jar</li> <li>struts-tiles-1.3.8.jar</li> <li>svnant-1.0.0.jar</li> <li>svnClientAdapter-1.0.0.jar</li> <li>wsdl4j-1.5.1.jar</li> <li>wss4j-1.5.0.jar</li> <li>xalan-2.7.0.jar</li> <li>xercesImpl-2.7.1.jar</li> <li>xmlsec-1.3.0.jar</li> </ul> <p>The dialog also shows a 'File name' field and a 'Files of type' dropdown set to '*.jar;*.zip'. 'Open' and 'Cancel' buttons are visible at the bottom right.</p>
18.	<p>Repeat <b>Step 17</b>, only now adding all the jar files in the <code>.../lib/generated</code> directory. These files are required so that the CPM services web pages display correctly. When finished, click <b>OK</b> to close the <b>Properties for clientsdk-webui</b> window.</p>

Step	Description
19.	<p>In BEA Weblogic 10.0, the <b>context.getRealPath()</b> method fails to return the <b>CPM-Client-Keystore</b> when WAR files are deployed. This results in the web application failing with a nullpointer error upon deployment. A <b>weblogic.xml</b> file is created to resolve this issue.</p> <p>Click the <b>File → New → File</b> menu to open the <b>New File</b> window. Expand the directory tree <b>project → WebContent</b> and select the <b>WEB-INF</b> folder. In the example below the <b>project</b> is <b>clientsdk-webui</b>. In the <b>File name:</b> text field enter <b>weblogic.xml</b>. Click <b>Finish</b> to create the file in the selected folder, and to return to the <b>BEA Workshop</b> window.</p> 

Step	Description
20.	<p>Double click the file created in the previous step in the <b>Project Explorer</b> window. The <b>weblogic.xml</b> file opens with the <b>weblogic.xml</b> tab in the BEA Workshop window. At the bottom of the <b>weblogic.xml</b> tab select the <b>Source</b> tab. This opens the file for adding source content to the file.</p>  <p>The screenshot shows the BEA Workshop for WebLogic Platform interface. The title bar reads 'J2EE - weblogic.xml - BEA Workshop for WebLogic Platform'. The menu bar includes File, Edit, Source, Navigate, Search, Project, Run, XML, Window, and Help. The toolbar contains various icons for file operations and development. The Project Explorer on the left shows a tree view for 'clientsdk-webui' with folders like 'Deployment Descriptor: CPMClient', 'Java Resources: src', 'build', 'WebContent', 'jsp_common', 'META-INF', and 'WEB-INF'. Under 'WEB-INF', the 'weblogic.xml' file is selected. The main editor window shows the 'weblogic.xml' file with the 'Source' tab active. The bottom status bar shows 'Writable', 'Smart Insert', and '1 : 1'. The Servers view at the bottom right shows 'BEA WebLogic Server v10.0 [wl]' with status 'Started' and 'clientsdk-webui' with status 'Republish'.</p>

Step	Description
------	-------------

21. Now add the following lines to the **weblogic.xml** file.

```

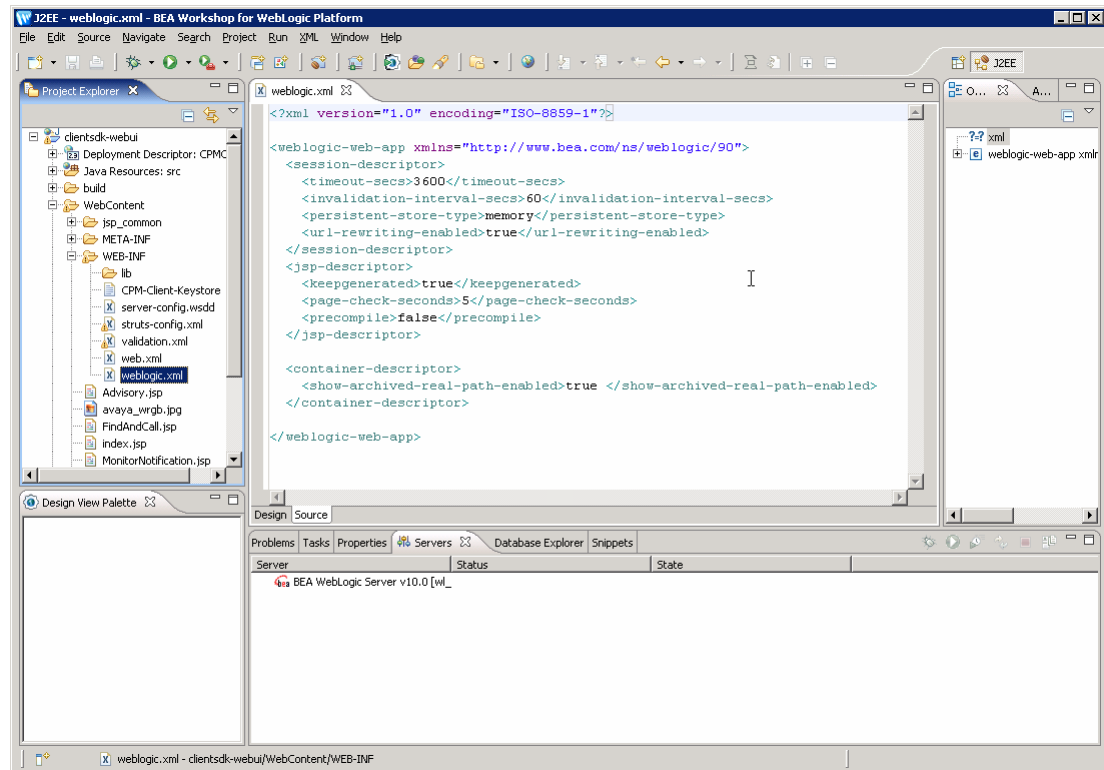
<?xml version="1.0" encoding="ISO-8859-1"?>

<weblogic-web-app xmlns="http://www.bea.com/ns/weblogic/90">
  <session-descriptor>
    <timeout-secs>3600</timeout-secs>
    <invalidation-interval-secs>60</invalidation-interval-secs>
    <persistent-store-type>memory</persistent-store-type>
    <url-rewriting-enabled>true</url-rewriting-enabled>
  </session-descriptor>
  <jsp-descriptor>
    <keepgenerated>true</keepgenerated>
    <page-check-seconds>5</page-check-seconds>
    <precompile>>false</precompile>
  </jsp-descriptor>

  <container-descriptor>
    <show-archived-real-path-enabled>true </show-archived-real-
path-enabled>
  </container-descriptor>
</weblogic-web-app>

```

The **container-descriptor** lines in the **weblogic.xml** file resolves the bug described in **step 19**. The following screenshot shows the resulting **weblogic.xml** file in BEA Workshop.

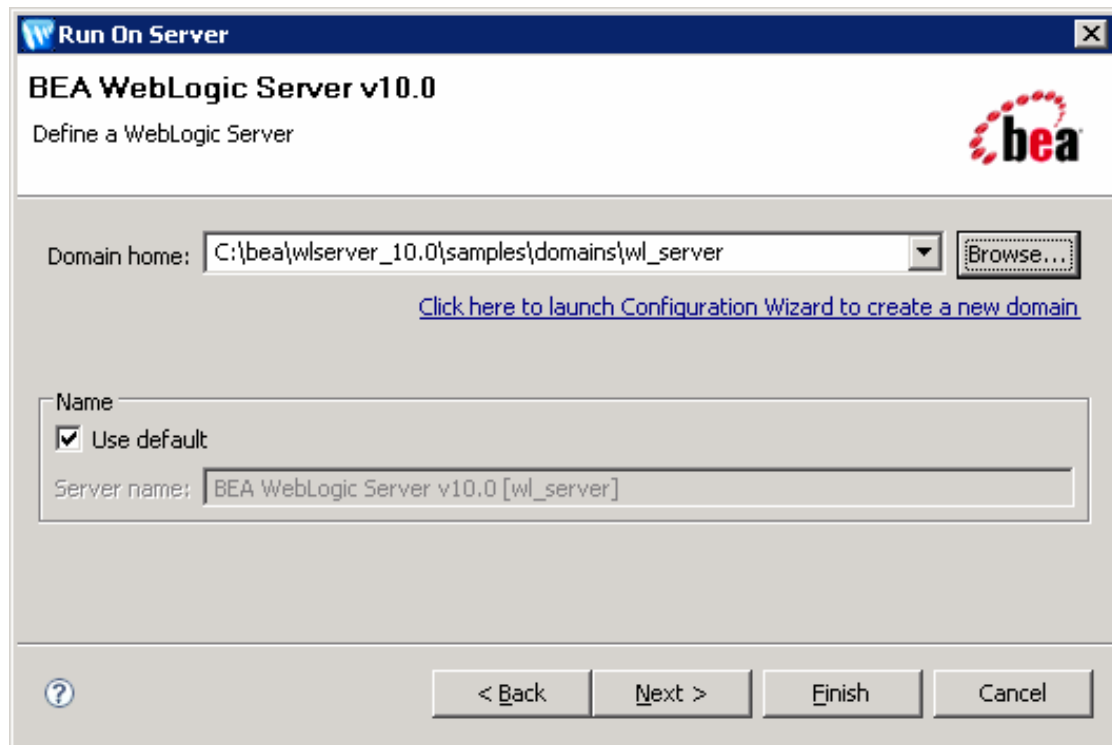


Step	Description
22.	<p>Save the <b>weblogic.xml</b> file by selecting the <b>File → Save</b> menu item (not shown). Force the project to recompile by selecting the <b>Project → Clean...</b> menu item. The <b>Clean</b> window opens (shown in <b>Step 15</b>). Select either radio button. If the <b>Clean projects selected below</b> radio button, then check the <b>clientsdk-webui</b> project in the list. Click the <b>OK</b> button to <b>clean</b> and force the project to rebuild. BEA Workshop is set to <b>build projects automatically</b>, therefore when <b>clean</b> is selected a rebuild also occurs.</p>

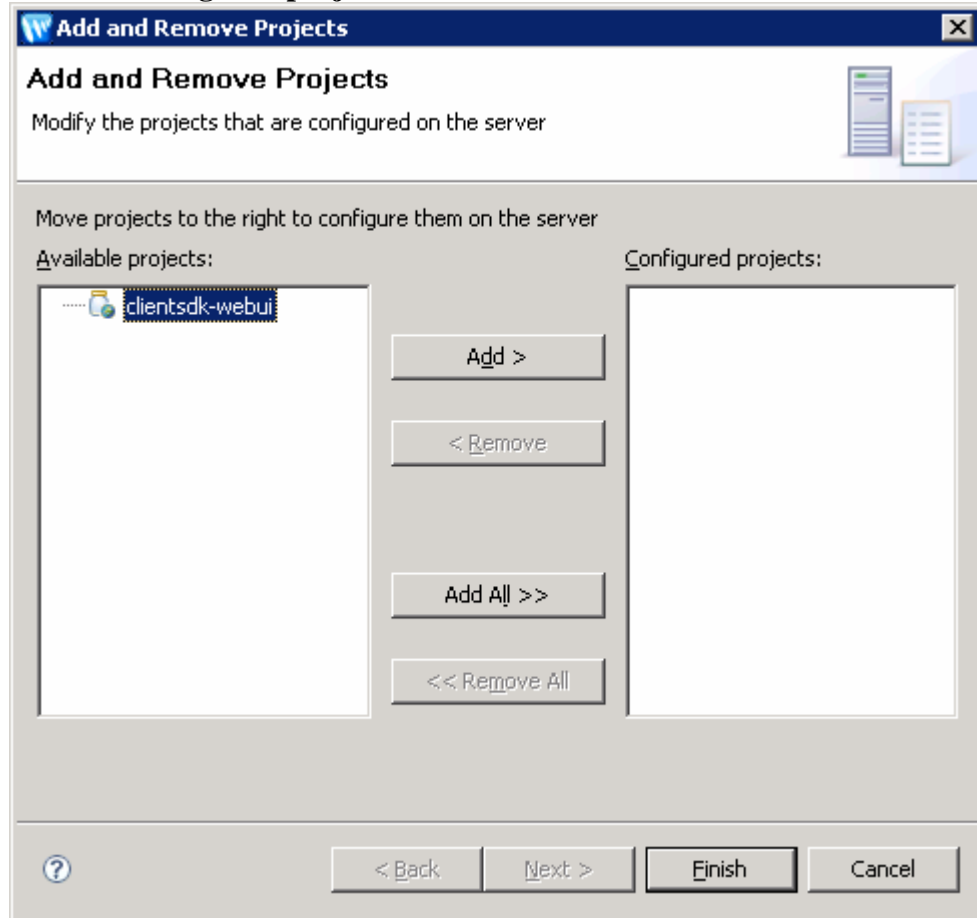
## 6. Deploy the Web Application via BEA Workshop

BEA Workshop includes the capability to deploy and test expanded directory web applications, from within the BEA Workshop environment. This capability allows a user to incrementally develop and test web applications. This section describes the steps for using this capability of BEA workshop.

1. Deploy the compiled web application to the BEA Weblogic server, to verify that the compiled application is valid. Right click on the **clientsdk-webui** project in the **Project Explorer** of **BEA Workshop**. Select **Run AS → Run on Server** in the drop down menu (not shown) that opens. The **Define a New Server** window opens (not shown). Accept the defaults and click **Next >**. The **BEA Weblogic Server v10.0** window opens. Click **Browse...** to open the **Browse For Folder** window. Browse to and select the **Domain Home** where you want to deploy the **clientsdk-webui** web application. The BEA samples (i.e., **wl\_server**) domain was used for these Application Notes.

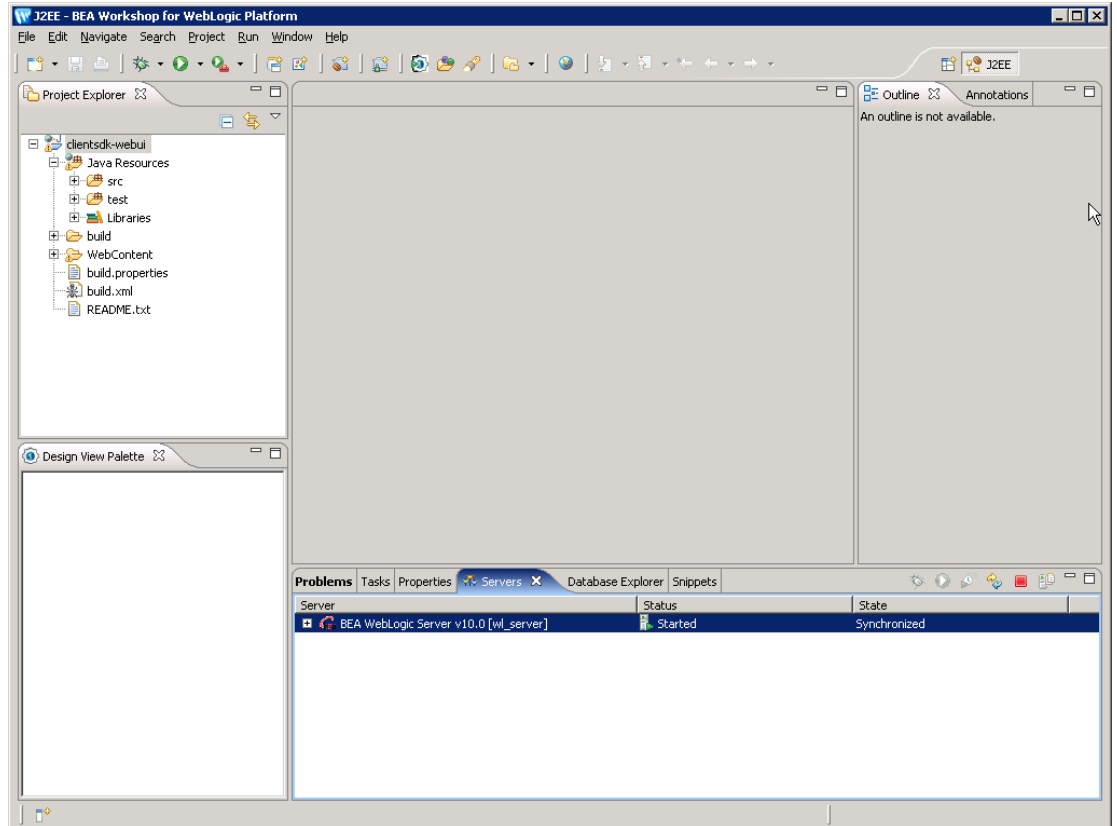



2. Click **Next >** to open the **Add and Remove Projects** window. Verify that the **clientsdk-webui** is in the **Configured projects:** frame of the window. If not, select the **clientsdk-webui** in the **Available projects:** frame, then click **Add** to add the project to the **Configured projects:** frame. Click **Next >**.

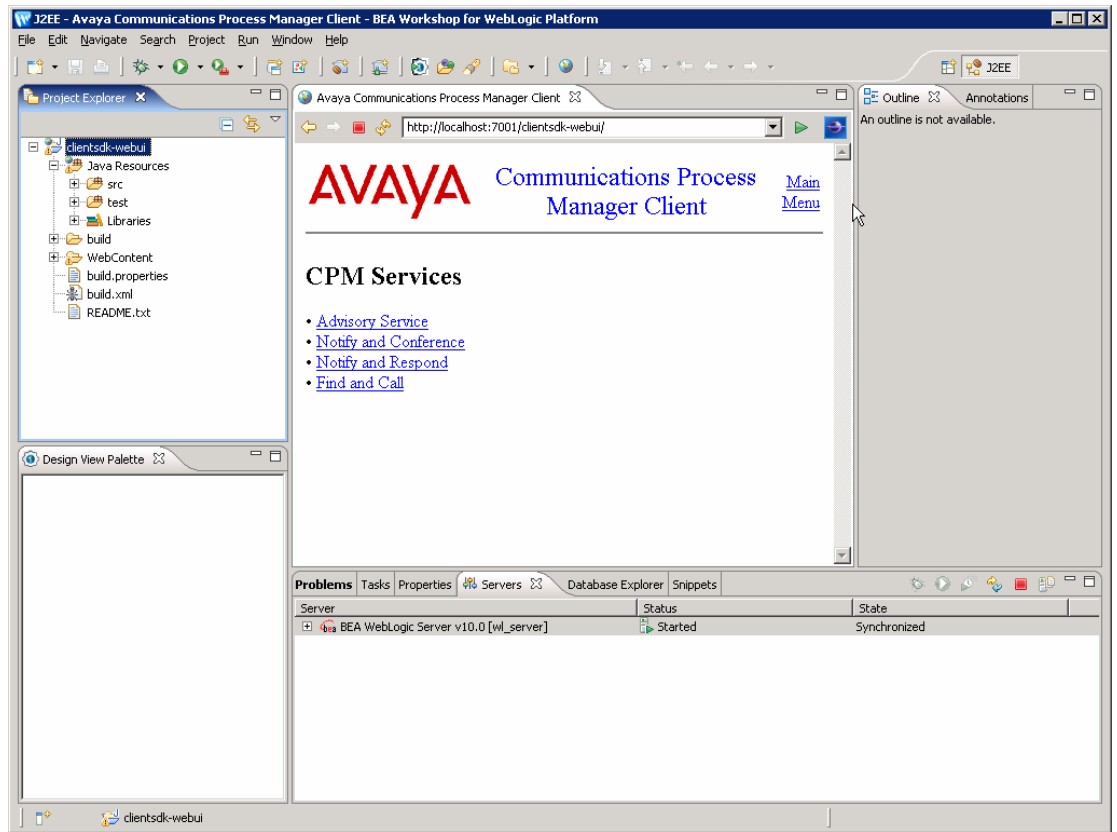


3. The **Selected Tasks** window opens (not shown). Leave as is and click **Finish**. If the Weblogic domain is not currently running it will startup and will display the **BEA WEBLOGIC SERVER 10.0 | Getting Started** page (not shown).

4. Go to the **Bea Workshop** window and select the **Servers** tab near the bottom of the window. The **Servers** tab should show the **Status** of the server as **Started**, and the **State** as **Synchronized**, once the server has started and completed the deployment of the project in expanded form.



5. Click the  icon in the tool bar at the top of the **BEA Workshop** window. This opens the **about.blank** tab (not shown), which is a blank browser web page. Type the url <http://localhost:7001/clientsdk-webui/> in the text box of the **about.blank** tab. The **Communications Process Manager Client** web page should open. The CPM SDK sample web application can now be verified.

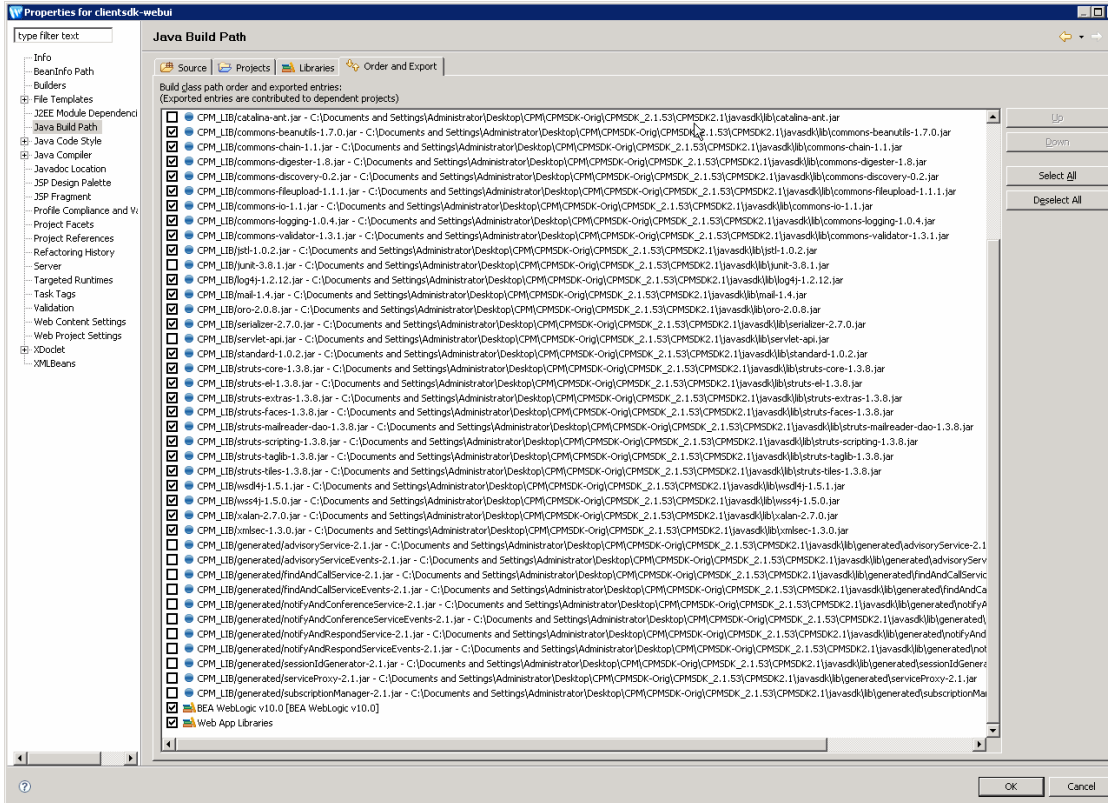


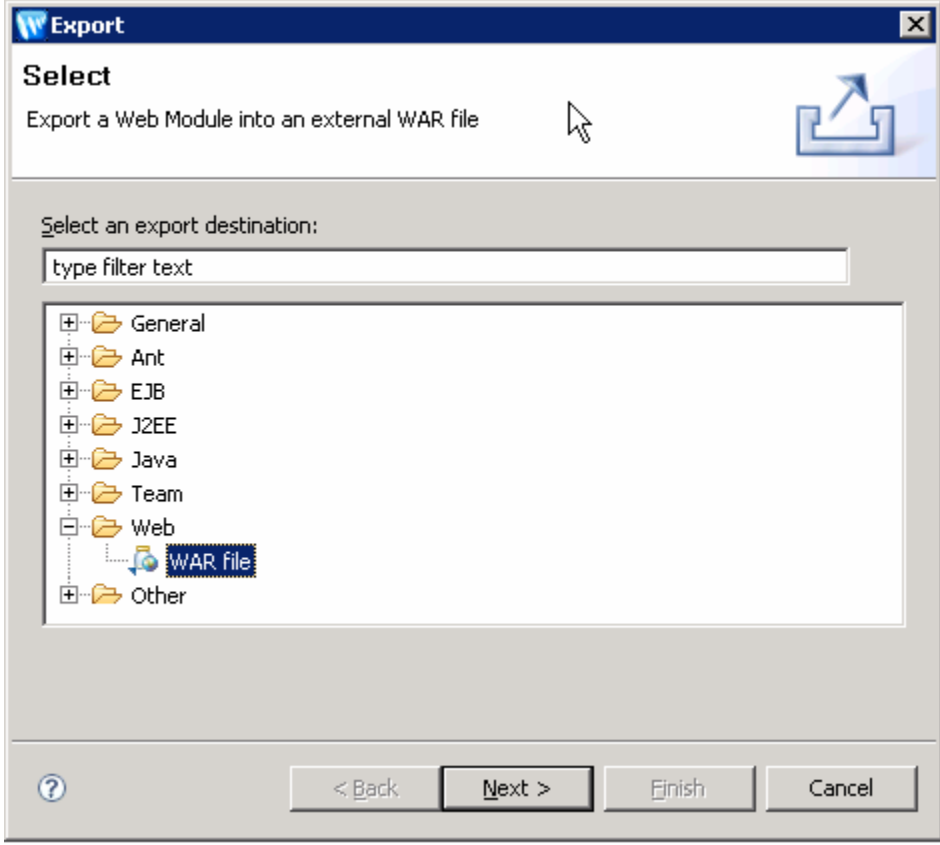
6. Steps 2 through 4 deployed the expanded directory for the **clientsdk-webui** web application to the Weblogic web server. The web application needs to be removed from the server, so that there are no conflicts when the WAR file is deployed to the server in **Section 7**.

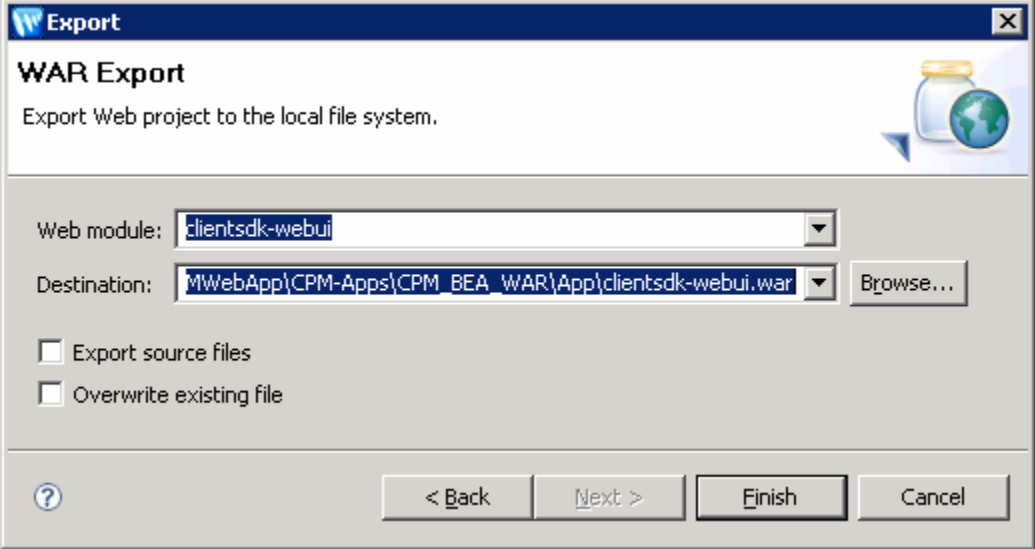
Right click on the running server in the **Servers** tab. In the drop menu select the **Add and Remove Projects...** menu to open the **Add and Remove Projects** window (shown in step 2 above). In the **Configured projects:** frame select the **clientsdk-webui** project. Click **< Remove** to move the project to the **Available projects:** frame. Then click **Finish**. This will remove the **clientsdk-webui** web application from the server.

# 7. Create Sample Avaya CPM Web Client Application WAR File

This section describes the steps used to generate a WAR file containing the CPM Web Client Application using the BEA Workshop export functions.

Step	Description
1.	<p>Right click the <b>clientsdk-webui</b> project in the <b>Project Explorer</b> tab of BEA Workshop. In the drop down menu that opens select the <b>Properties</b> menu item (not shown). In the left frame of the <b>Properties for clientsdk-webui</b> select the <b>Java Build Path</b> item. The <b>Java Build Path</b> frame will open in the right frame. Select the <b>Order and Export</b> tab, in the <b>Java Build Path</b> frame. Click the <b>Select All</b> button. Then uncheck the <b>Catalina-ant.jar</b>, <b>junit-3.8.jar</b>, <b>servlet-api.jar</b>, and all the <b>.../generated/*.jar</b> checkboxes. Then click <b>OK</b>.</p> 

Step	Description
2.	<p>Now export the <b>clientsdk-webui</b> project to a WAR file. Select the <b>File → Export...</b> menu item to open the <b>Select</b> window. Expand the <b>Web</b> folder, and select the <b>WAR File</b> item, as shown below. Click <b>Next</b>.</p> 

Step	Description
3.	<p>The <b>WAR Export</b> window opens. In the <b>Web module</b> field enter <b>clientsdk-webui</b>. Click the '<b>Browse...</b>' button and browse to the directory where the WAR file will be saved. Remember this location, since it will be used for deployment of the WAR file on the BEA Weblogic Application Server. Click <b>Finish</b>. The WAR file is saved to the specified location.</p> 

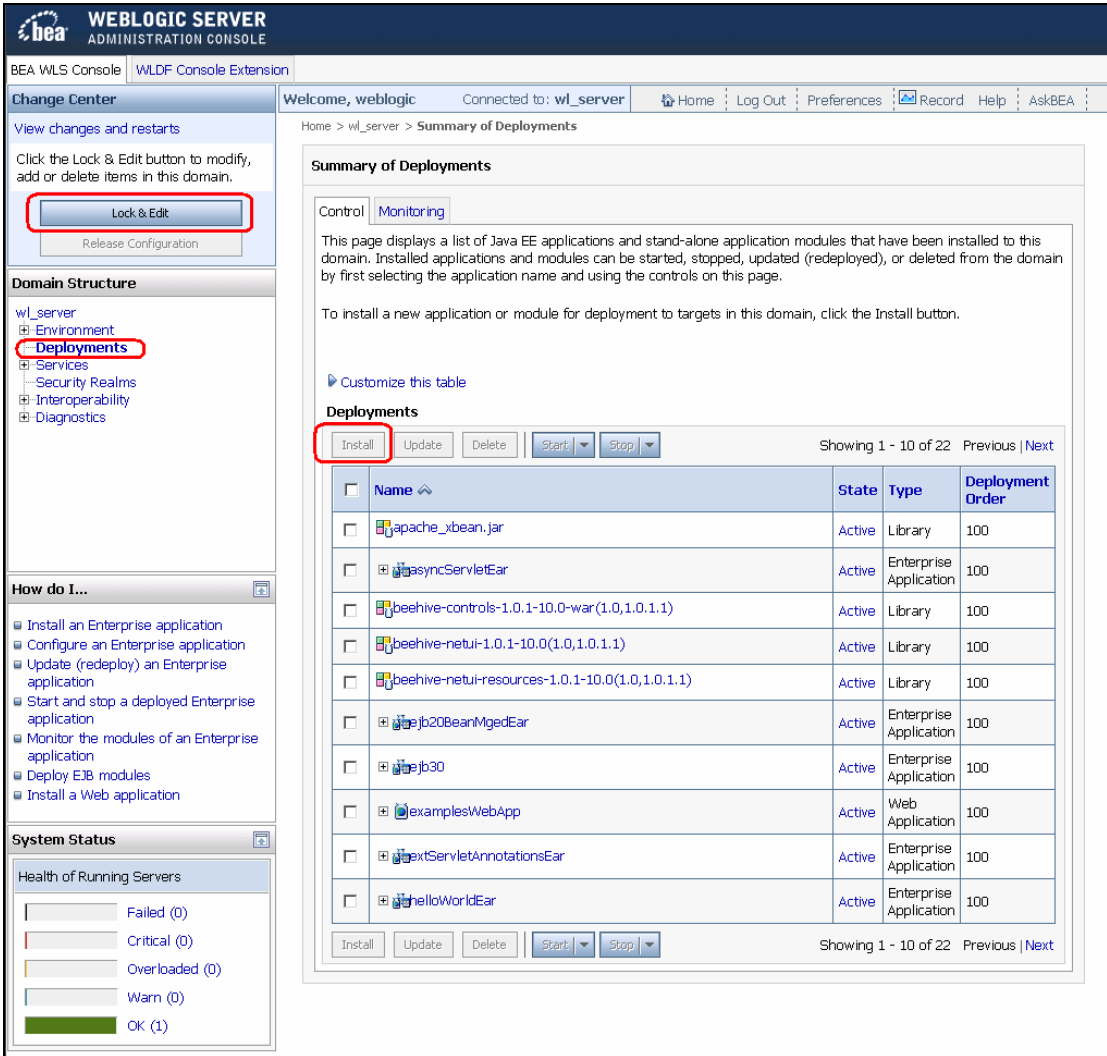
## 8. Deploy Sample Avaya CPM Web Client Application to BEA Weblogic Application Server

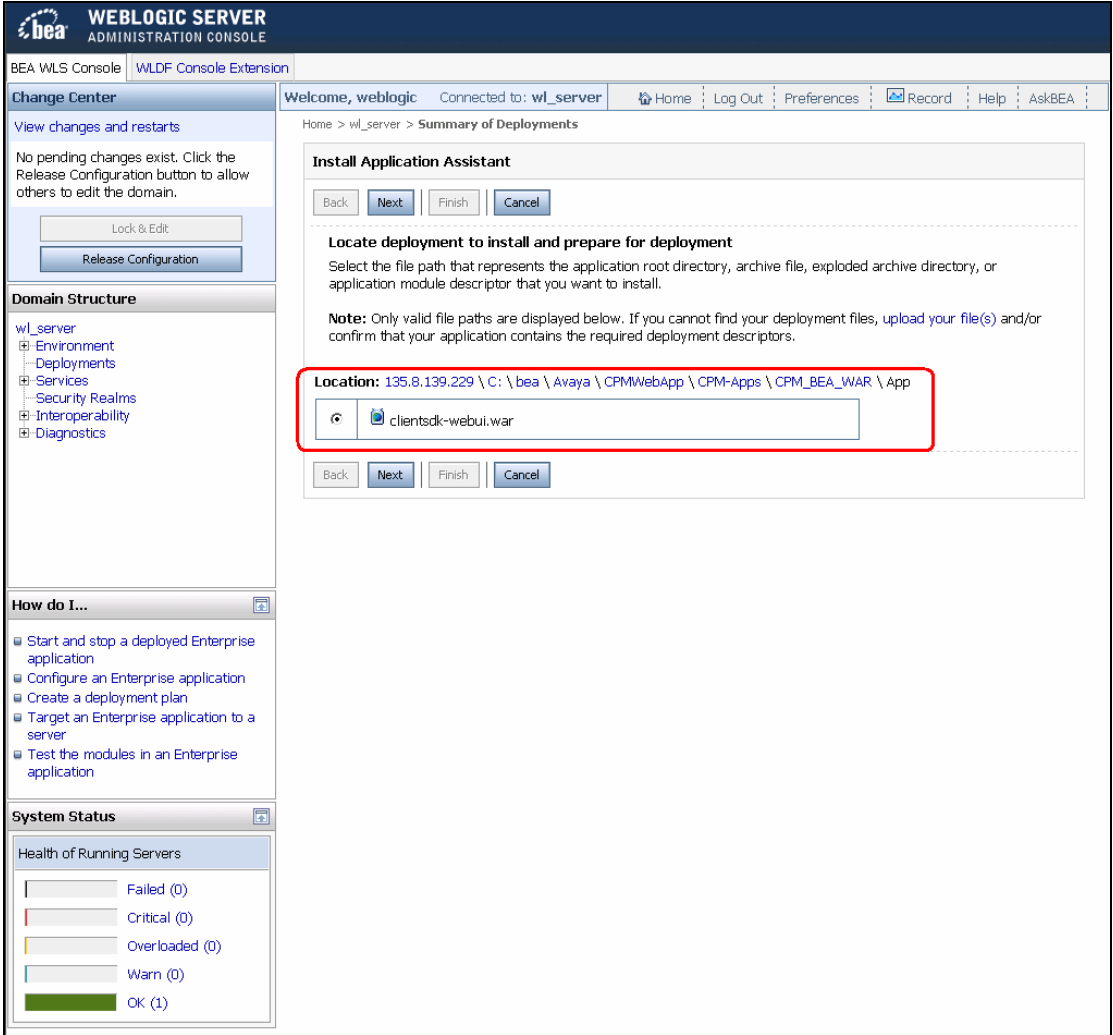
These Application Notes assume that the BEA Weblogic Application Server is already installed on Microsoft Windows XP. Refer to [9] - [14] in **Section 11** for additional information on the installation of BEA Weblogic Application Server.

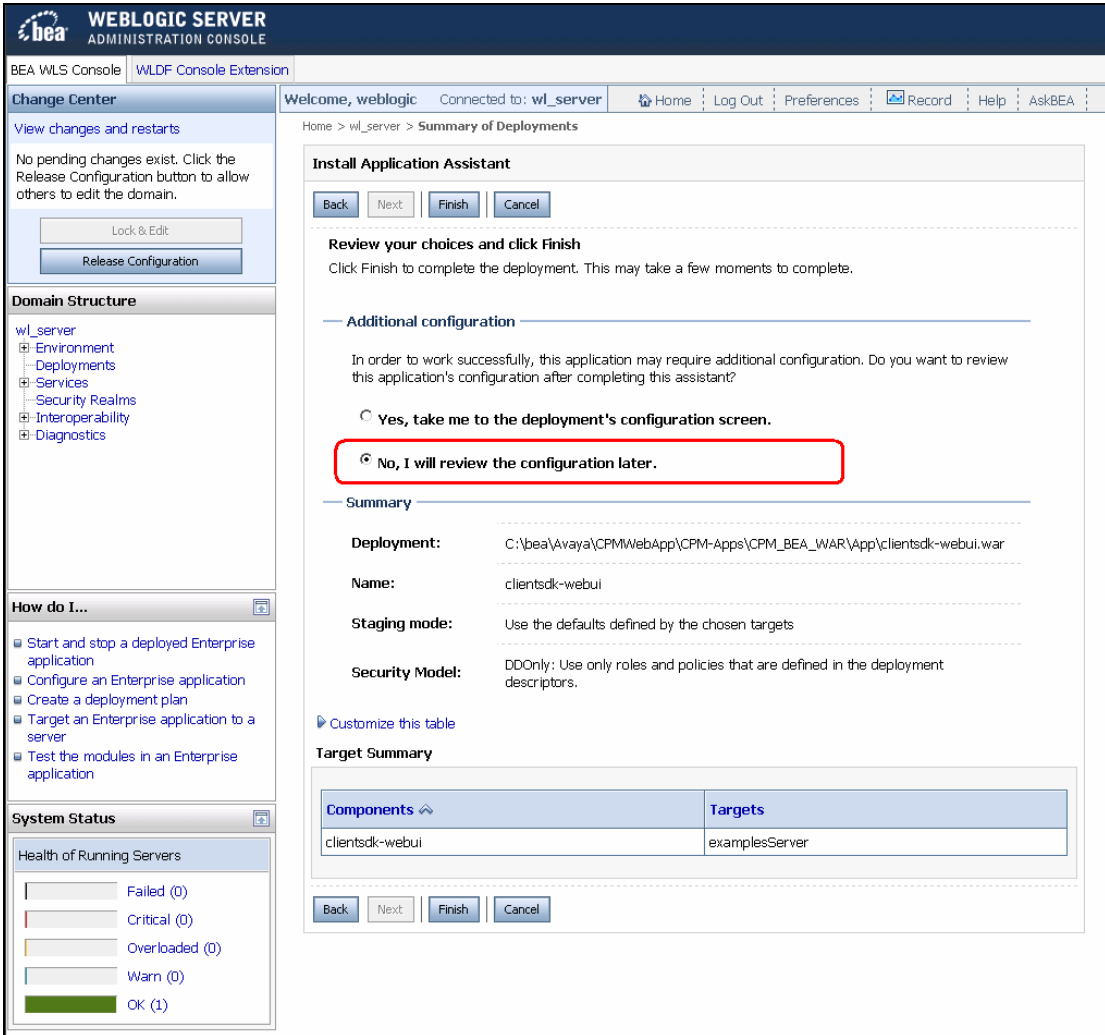
This section describes the steps used to deploy the CPM SDK client web application as a WAR file to a running BEA Weblogic Application Server domain.

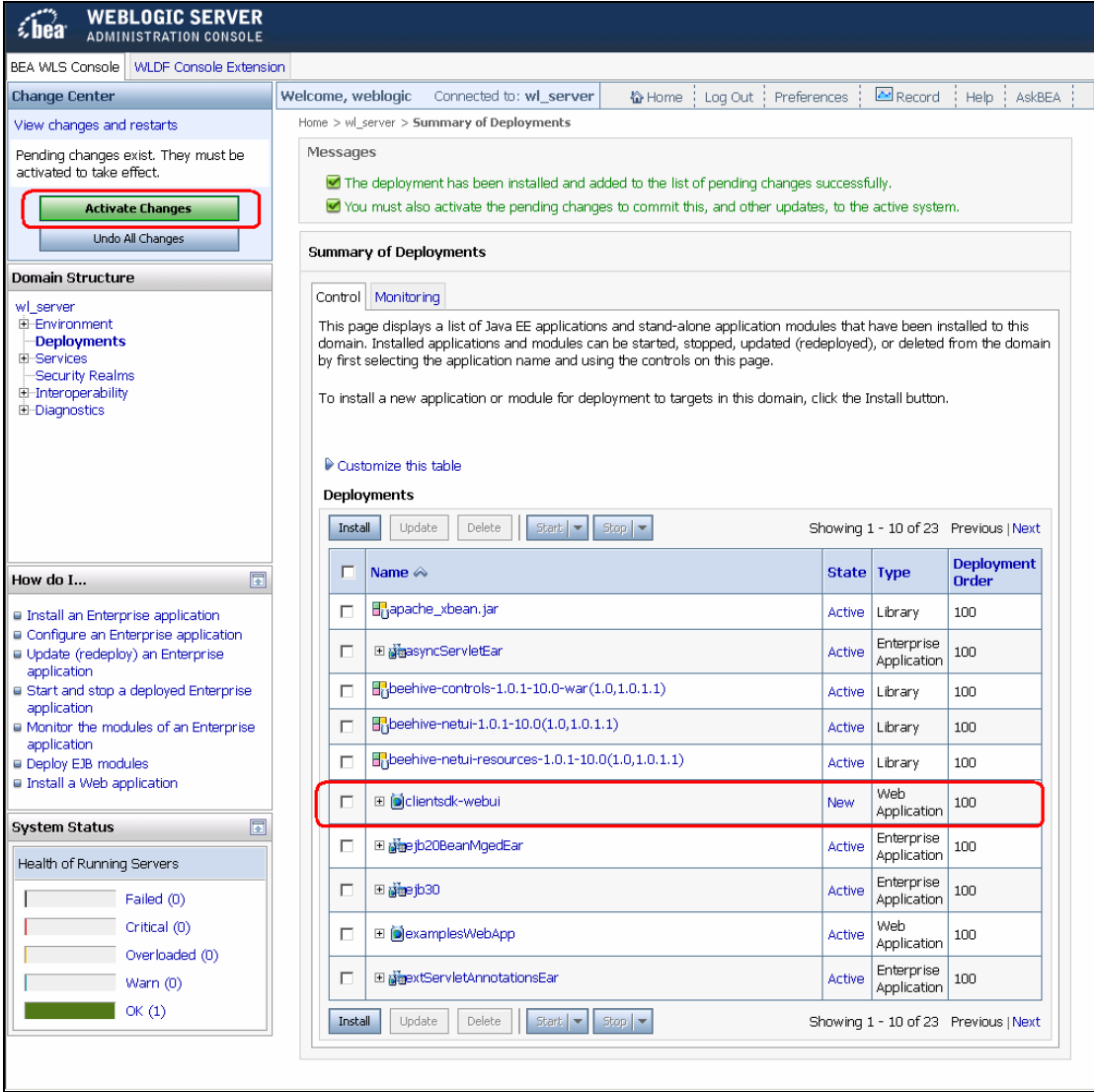
In order to deploy a web application to a BEA Weblogic domain, the deployment file(s) must be accessible to the domain's administration server. For these application notes a single server environment is used. That is, the domain and administration server are on the same system.

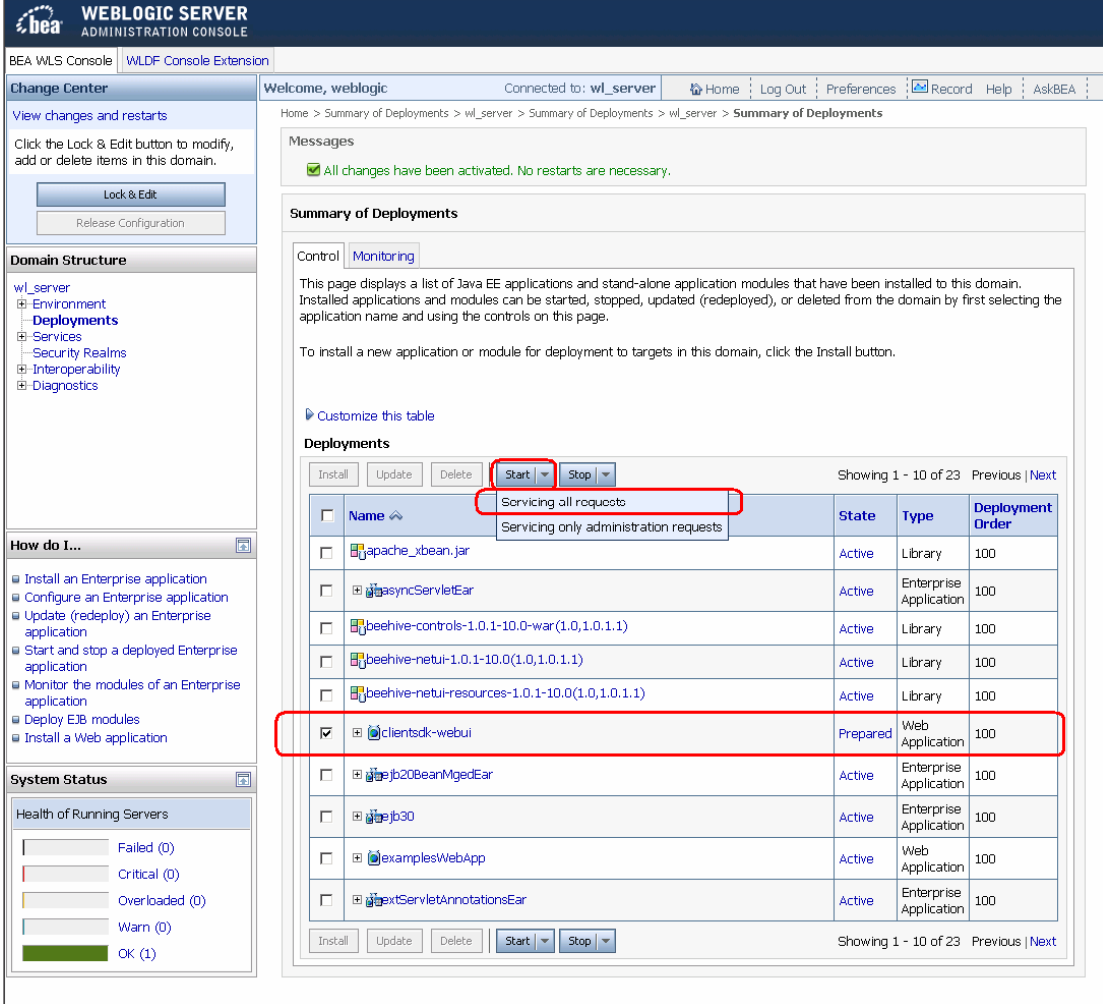
Step	Description
1.	Copy the clienksdk-webui.war file to the installation directory of the BEA Weblogic domain's Administration server. The installation directory is any directory on the administration server created for storing and tracking applications for deployment.
2.	Log into BEA Weblogic application server console interface by opening a web browser to <b>http://&lt;BEA Weblogic IP Address&gt;:7001/console</b> . The BEA Weblogic Server administration console login screen opens (not shown). Log in with the appropriate credentials.

Step	Description																																												
3.	<p>The BEA Weblogic <b>ADMINISTRATION CONSOLE</b> window opens. Select the <b>Deployments</b> item in the <b>Domain Structure</b> frame. The <b>Summary of Deployments</b> frame opens on the right. In the <b>Change Center</b> frame click the <b>Lock &amp; Edit</b> button. This enables the <b>Install</b> button in the <b>Summary of Deployments</b> frame. Click the <b>Install</b> button.</p>  <p>The screenshot shows the BEA Weblogic Administration Console interface. On the left, the 'Change Center' panel has a 'Lock &amp; Edit' button highlighted with a red box. Below it, the 'Domain Structure' tree shows 'Deployments' highlighted with a red box. At the bottom left, the 'System Status' panel shows 'OK (1)' servers. The main content area displays the 'Summary of Deployments' page, which includes a table of installed applications. The 'Install' button in the table's control bar is highlighted with a red box.</p> <table border="1" data-bbox="617 877 1380 1375"> <thead> <tr> <th>Name</th> <th>State</th> <th>Type</th> <th>Deployment Order</th> </tr> </thead> <tbody> <tr> <td>apache_xbean.jar</td> <td>Active</td> <td>Library</td> <td>100</td> </tr> <tr> <td>asyncServletEar</td> <td>Active</td> <td>Enterprise Application</td> <td>100</td> </tr> <tr> <td>beehive-controls-1.0.1-10.0-war(1.0,1.0.1.1)</td> <td>Active</td> <td>Library</td> <td>100</td> </tr> <tr> <td>beehive-netui-1.0.1-10.0(1.0,1.0.1.1)</td> <td>Active</td> <td>Library</td> <td>100</td> </tr> <tr> <td>beehive-netui-resources-1.0.1-10.0(1.0,1.0.1.1)</td> <td>Active</td> <td>Library</td> <td>100</td> </tr> <tr> <td>jb20BeanMgedEar</td> <td>Active</td> <td>Enterprise Application</td> <td>100</td> </tr> <tr> <td>jb30</td> <td>Active</td> <td>Enterprise Application</td> <td>100</td> </tr> <tr> <td>examplesWebApp</td> <td>Active</td> <td>Web Application</td> <td>100</td> </tr> <tr> <td>textServletAnnotationsEar</td> <td>Active</td> <td>Enterprise Application</td> <td>100</td> </tr> <tr> <td>helloWorldEar</td> <td>Active</td> <td>Enterprise Application</td> <td>100</td> </tr> </tbody> </table>	Name	State	Type	Deployment Order	apache_xbean.jar	Active	Library	100	asyncServletEar	Active	Enterprise Application	100	beehive-controls-1.0.1-10.0-war(1.0,1.0.1.1)	Active	Library	100	beehive-netui-1.0.1-10.0(1.0,1.0.1.1)	Active	Library	100	beehive-netui-resources-1.0.1-10.0(1.0,1.0.1.1)	Active	Library	100	jb20BeanMgedEar	Active	Enterprise Application	100	jb30	Active	Enterprise Application	100	examplesWebApp	Active	Web Application	100	textServletAnnotationsEar	Active	Enterprise Application	100	helloWorldEar	Active	Enterprise Application	100
Name	State	Type	Deployment Order																																										
apache_xbean.jar	Active	Library	100																																										
asyncServletEar	Active	Enterprise Application	100																																										
beehive-controls-1.0.1-10.0-war(1.0,1.0.1.1)	Active	Library	100																																										
beehive-netui-1.0.1-10.0(1.0,1.0.1.1)	Active	Library	100																																										
beehive-netui-resources-1.0.1-10.0(1.0,1.0.1.1)	Active	Library	100																																										
jb20BeanMgedEar	Active	Enterprise Application	100																																										
jb30	Active	Enterprise Application	100																																										
examplesWebApp	Active	Web Application	100																																										
textServletAnnotationsEar	Active	Enterprise Application	100																																										
helloWorldEar	Active	Enterprise Application	100																																										

Step	Description
4.	<p>The <b>Install Application Assistant</b> frame opens in the right frame. Navigate to the saved WAR file (see <b>Step 3 in Section 7</b> above), by clicking the folder in the <b>Location</b> area of this frame. Select the <b>clientsdk-webui.war</b> radio button to select the WAR file for deployment. Click the <b>Next</b> button.</p>  <p>The screenshot shows the BEA Weblogic Server Administration Console interface. The main content area displays the 'Install Application Assistant' dialog box. The dialog has a title bar with 'Back', 'Next', 'Finish', and 'Cancel' buttons. Below the title bar, there is a section titled 'Locate deployment to install and prepare for deployment'. It contains a text area with the path 'Location: 135.8.139.229 \ C: \ bea \ Avaya \ CPMWebApp \ CPM-Apps \ CPM_BEA_WAR \ App' and a file selection area where 'clientsdk-webui.war' is selected. A red box highlights this selection area. Below the file selection area, there are 'Back', 'Next', 'Finish', and 'Cancel' buttons. On the left side of the console, there is a 'Change Center' section with 'View changes and restarts' and a 'Release Configuration' button. Below that is a 'Domain Structure' tree showing 'wl_server' and its sub-items. At the bottom, there is a 'System Status' section showing 'Health of Running Servers' with a bar chart indicating 'OK (1)'.</p>
5.	<p>The <b>Install Application Assistant</b> frame updates showing the <b>Choose targeting style</b> selections (not shown). Leave as default and click <b>Next</b>.</p>
6.	<p>The <b>Install Application Assistant</b> frame updates showing the <b>Options Settings</b> selections (not shown). Leave as default and click <b>Next</b>.</p>

Step	Description				
7.	<p>The <b>Install Application Assistant</b> frame updates showing the <b>Review your choices and click Finish</b> selections. Select the <b>No, I will review the configuration later</b> radio button, in the <b>Additional configuration</b> area of the frame. Then click <b>Finish</b>. Note the name of the web application in the <b>Components</b> column of the table at the bottom of the frame. This is the URL used to open the application in a web browser.</p>  <p>The screenshot displays the BEA WebLogic Server Administration Console interface. The main content area shows the 'Install Application Assistant' dialog box. At the top of the dialog, there are buttons for 'Back', 'Next', 'Finish', and 'Cancel'. Below these, the text reads 'Review your choices and click Finish' and 'Click Finish to complete the deployment. This may take a few moments to complete.' The 'Additional configuration' section asks, 'In order to work successfully, this application may require additional configuration. Do you want to review this application's configuration after completing this assistant?' There are two radio buttons: 'Yes, take me to the deployment's configuration screen.' and 'No, I will review the configuration later.' The second radio button is selected and highlighted with a red rectangular box. Below this is a 'Summary' section with the following details:     <ul style="list-style-type: none"> <li><b>Deployment:</b> C:\bea\Avaya\CPMWebApp\CPM-Apps\CPM_BEA_WAR\App\clientsdk-webui.war</li> <li><b>Name:</b> clientsdk-webui</li> <li><b>Staging mode:</b> Use the defaults defined by the chosen targets</li> <li><b>Security Model:</b> DDOnly: Use only roles and policies that are defined in the deployment descriptors.</li> </ul>     At the bottom of the dialog is a 'Target Summary' table:     <table border="1" data-bbox="625 1218 1380 1312"> <thead> <tr> <th>Components</th> <th>Targets</th> </tr> </thead> <tbody> <tr> <td>clientsdk-webui</td> <td>examplesServer</td> </tr> </tbody> </table>     The left sidebar of the console shows the 'Domain Structure' for 'wl_server' with a tree view including Environment, Deployments, Services, Security Realms, Interoperability, and Diagnostics. The 'System Status' section shows the health of running servers: Failed (0), Critical (0), Overloaded (0), Warn (0), and OK (1).           </p>	Components	Targets	clientsdk-webui	examplesServer
Components	Targets				
clientsdk-webui	examplesServer				

Step	Description
8.	<p>The <b>Summary of Deployment</b> frame opens. The <b>clientsdk-webui</b> web Application will be listed in the <b>Deployments</b> table and the state will be <b>New</b>. Also, the <b>Lock &amp; Edit</b> button changes to the <b>Activate Changes</b> button in the <b>Change Center</b> frame. Click the <b>Activate Changes</b> button.</p>  <p>The screenshot shows the BEA WebLogic Server Administration Console interface. The main content area displays the 'Summary of Deployments' page. On the left, the 'Change Center' panel shows a green 'Activate Changes' button, which is highlighted with a red rectangular box. Below it, the 'Domain Structure' tree is visible, showing the 'wls_server' domain with various sub-entities like 'Environment', 'Deployments', 'Services', etc. The 'Deployments' table in the main area lists several applications, with 'clientsdk-webui' highlighted by a red box. The table has columns for 'Name', 'State', 'Type', and 'Deployment Order'. The 'clientsdk-webui' entry shows a state of 'New' and a type of 'Web Application'. Other entries include 'apache_xbean.jar', 'beehive-controls-1.0.1-10.0-war(1.0,1.0.1.1)', 'beehive-netui-1.0.1-10.0(1.0,1.0.1.1)', 'beehive-netui-resources-1.0.1-10.0(1.0,1.0.1.1)', 'jb20BeanMgedEar', 'jb30', 'examplesWebApp', and 'ServletAnnotationsEar'. The 'State' column for 'clientsdk-webui' is 'New', while others are 'Active'. The 'Deployment Order' is 100 for all entries.</p>

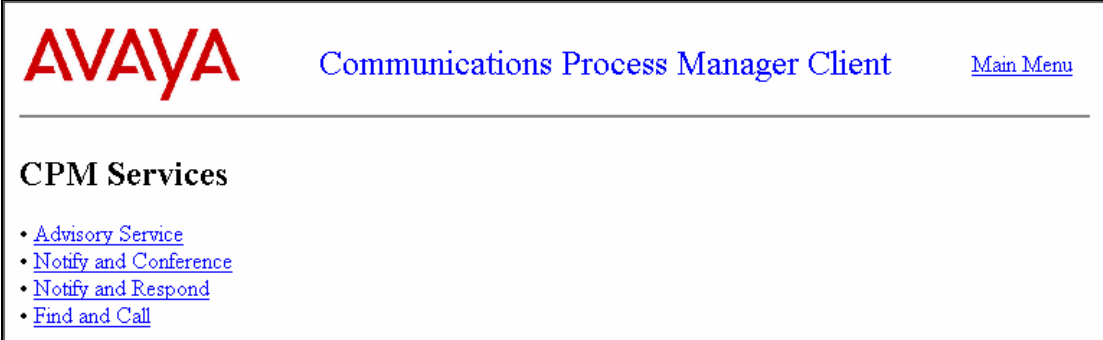
Step	Description																																												
9.	<p>The <b>Summary of Deployments</b> frame updates, and the web application state changes to <b>Prepared</b>. Check the <b>clientsdk-webui</b> checkbox in the <b>Deployments</b> table. Then click the <b>Start</b> button, and select the <b>Serving all requests</b> menu choice.</p>  <p>The screenshot shows the WebLogic Server Administration Console interface. The main content area displays the 'Summary of Deployments' page. On the left, there is a 'Domain Structure' tree with 'Deployments' selected. Below it is a 'How do I...' section with various tasks. At the bottom left is a 'System Status' section showing the health of running servers. The main table lists several deployments, with 'clientsdk-webui' highlighted in red. Its state is 'Prepared'. Above the table, the 'Start' button and its dropdown menu are highlighted in red, with 'Serving all requests' selected. The table has columns for Name, State, Type, and Deployment Order.</p> <table border="1" data-bbox="609 840 1372 1312"> <thead> <tr> <th>Name</th> <th>State</th> <th>Type</th> <th>Deployment Order</th> </tr> </thead> <tbody> <tr> <td>clientsdk-webui</td> <td>Prepared</td> <td>Web Application</td> <td>100</td> </tr> <tr> <td>apache_xbean.jar</td> <td>Active</td> <td>Library</td> <td>100</td> </tr> <tr> <td>asyncServletEar</td> <td>Active</td> <td>Enterprise Application</td> <td>100</td> </tr> <tr> <td>beehive-controls-1.0.1-10.0-war(1.0,1.0.1.1)</td> <td>Active</td> <td>Library</td> <td>100</td> </tr> <tr> <td>beehive-netui-1.0.1-10.0(1.0,1.0.1.1)</td> <td>Active</td> <td>Library</td> <td>100</td> </tr> <tr> <td>beehive-netui-resources-1.0.1-10.0(1.0,1.0.1.1)</td> <td>Active</td> <td>Library</td> <td>100</td> </tr> <tr> <td>ejb20BeanMgedEar</td> <td>Active</td> <td>Enterprise Application</td> <td>100</td> </tr> <tr> <td>ejb30</td> <td>Active</td> <td>Enterprise Application</td> <td>100</td> </tr> <tr> <td>examplesWebApp</td> <td>Active</td> <td>Web Application</td> <td>100</td> </tr> <tr> <td>extServletAnnotationsEar</td> <td>Active</td> <td>Enterprise Application</td> <td>100</td> </tr> </tbody> </table>	Name	State	Type	Deployment Order	clientsdk-webui	Prepared	Web Application	100	apache_xbean.jar	Active	Library	100	asyncServletEar	Active	Enterprise Application	100	beehive-controls-1.0.1-10.0-war(1.0,1.0.1.1)	Active	Library	100	beehive-netui-1.0.1-10.0(1.0,1.0.1.1)	Active	Library	100	beehive-netui-resources-1.0.1-10.0(1.0,1.0.1.1)	Active	Library	100	ejb20BeanMgedEar	Active	Enterprise Application	100	ejb30	Active	Enterprise Application	100	examplesWebApp	Active	Web Application	100	extServletAnnotationsEar	Active	Enterprise Application	100
Name	State	Type	Deployment Order																																										
clientsdk-webui	Prepared	Web Application	100																																										
apache_xbean.jar	Active	Library	100																																										
asyncServletEar	Active	Enterprise Application	100																																										
beehive-controls-1.0.1-10.0-war(1.0,1.0.1.1)	Active	Library	100																																										
beehive-netui-1.0.1-10.0(1.0,1.0.1.1)	Active	Library	100																																										
beehive-netui-resources-1.0.1-10.0(1.0,1.0.1.1)	Active	Library	100																																										
ejb20BeanMgedEar	Active	Enterprise Application	100																																										
ejb30	Active	Enterprise Application	100																																										
examplesWebApp	Active	Web Application	100																																										
extServletAnnotationsEar	Active	Enterprise Application	100																																										

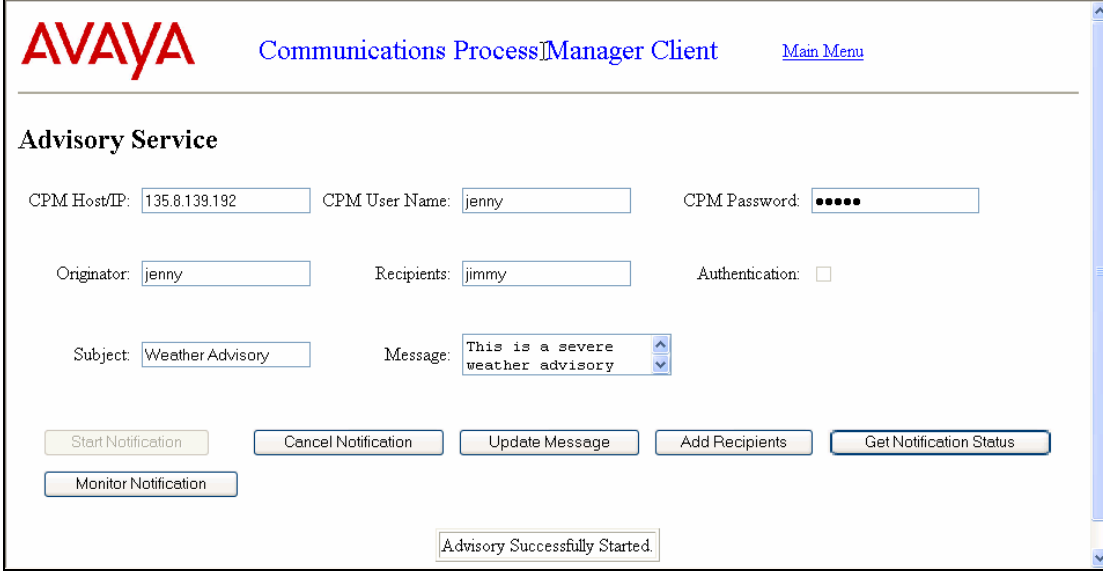
Step	Description
10.	<p>The <b>Start Application Assistant</b> frame opens (not shown). Click the <b>Yes</b> button, to start the <b>clientsdk-webui</b> application. The <b>Summary of Deployments</b> frame updates, and there should be no errors displayed in the <b>Messages</b> area at the top of this frame. The <b>clientsdk-webui</b> state should now be <b>Active</b>.</p> <p>The screenshot shows the BEA Weblogic Server Administration Console interface. The main content area is titled 'Summary of Deployments'. At the top, there is a 'Messages' section with a green checkmark and the text: 'Start requests have been sent to the selected Deployments.' Below this is a table of deployments. The table has columns for 'Name', 'State', 'Type', and 'Deployment Order'. The 'clientsdk-webui' application is highlighted with a red box, showing a state of 'Active' and a type of 'Web Application'. Other applications listed include 'apache_xbean.jar', 'asyncServletEar', 'beehive-controls-1.0.1-10.0-war(1.0,1.0.1.1)', 'beehive-netui-1.0.1-10.0(1.0,1.0.1.1)', 'beehive-netui-resources-1.0.1-10.0(1.0,1.0.1.1)', 'jb20BeanMgedEar', 'jb30', 'examplesWebApp', and 'textServletAnnotationsEar'. The left sidebar contains navigation options like 'Change Center', 'Domain Structure', 'How do I...', and 'System Status'.</p>

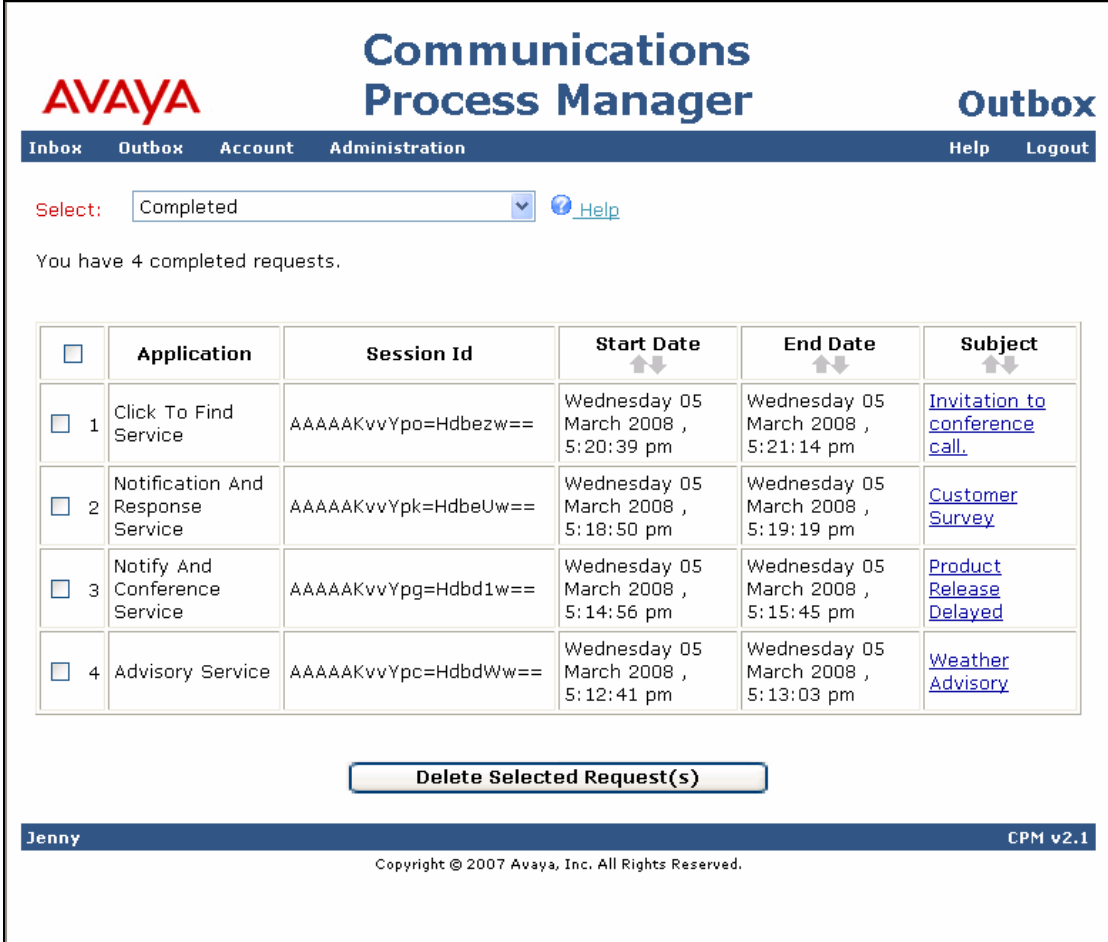
## 9. Verification Steps

These Application Notes were confirmed using the steps below to verify that the sample Avaya CPM Client web Client was deployed correctly, and was operational on the BEA Weblogic Application Server.

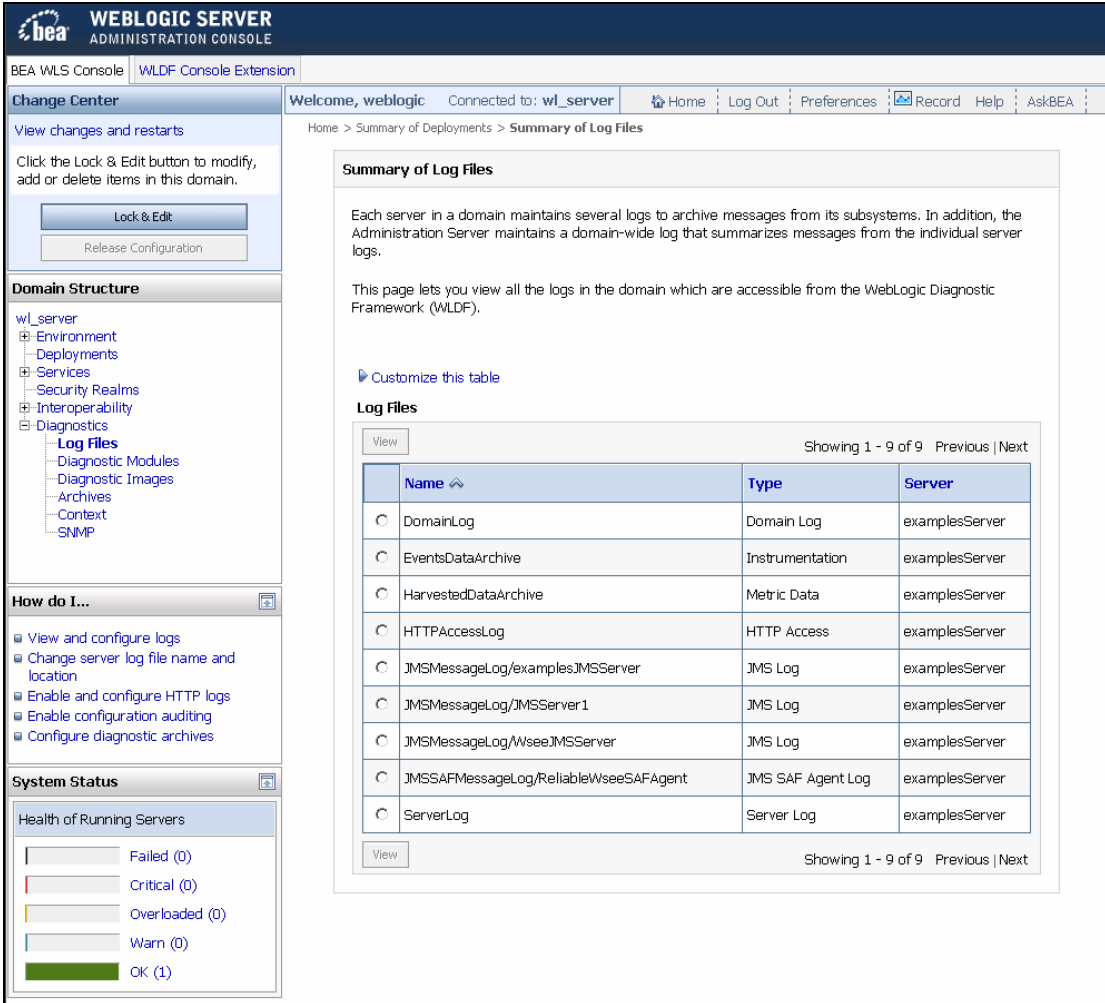
- Verify that the sample Avaya CPM Web Client application is deployed to the BEA Weblogic Application Server.
- Verify that the CPM Advisory service can be launched and acknowledged by all the recipients.
- Verify that CPM Notify and Conference service can be launched and the recipients can join the conference.
- Verify that CPM Notify and Respond service can be launched and the recipients can respond to the notification.
- Verify that CPM Find and Call service can be launched and conference the recipients.
- Verify Avaya CPM & BEA Weblogic log files do not show any errors.

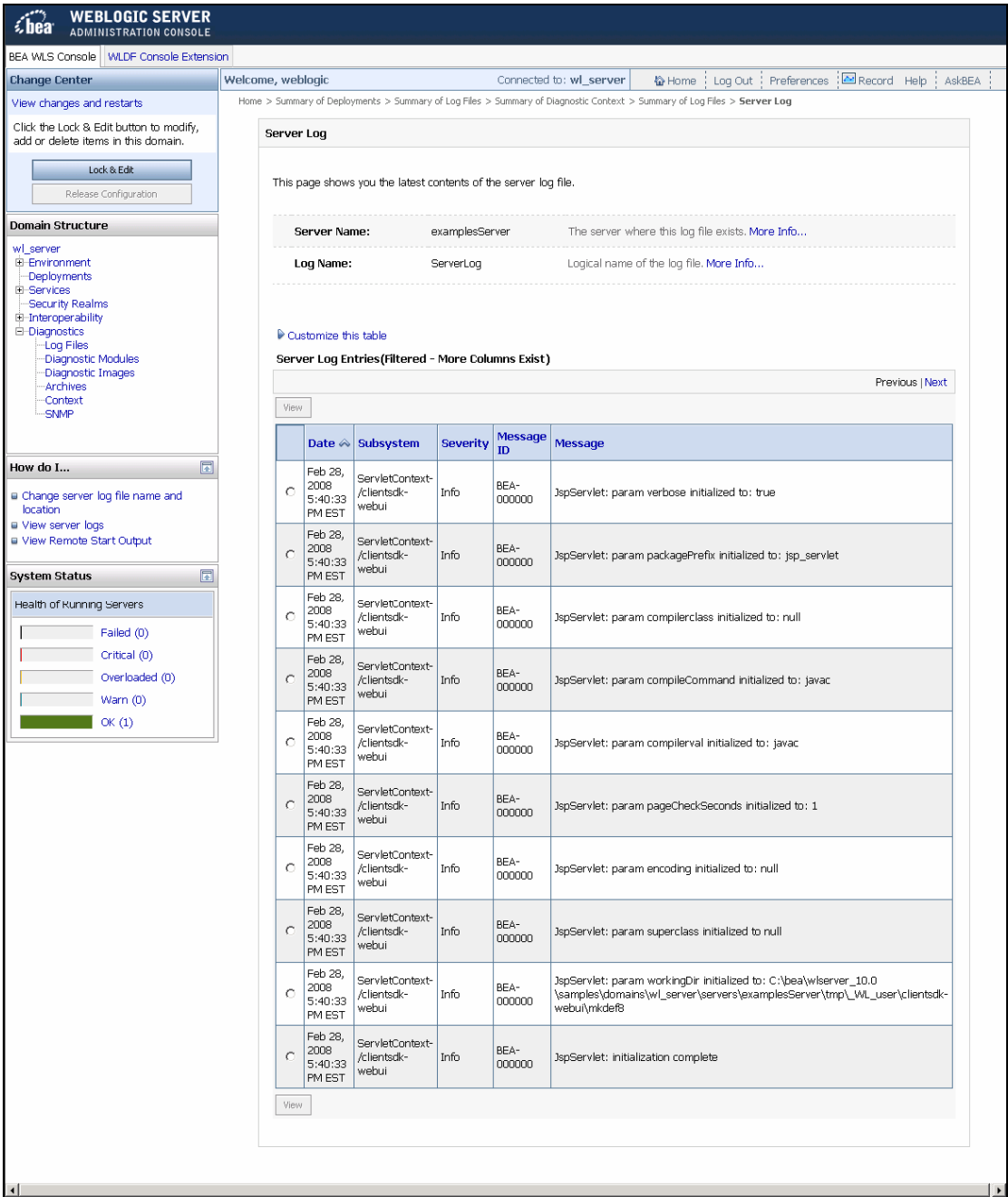
Step	Description
1.	<p>Verify that the CPMClient is deployed to the BEA Weblogic Application Server. Launch a web browser, and enter the URL <b>http://&lt; BEA Weblogic AS server IP address&gt;:7001/clientsdk-webui</b>. Verify that the main <b>Communications Process Manager Client</b> web page opens.</p> <div data-bbox="318 877 1414 1213" style="border: 1px solid black; padding: 10px;">  </div>

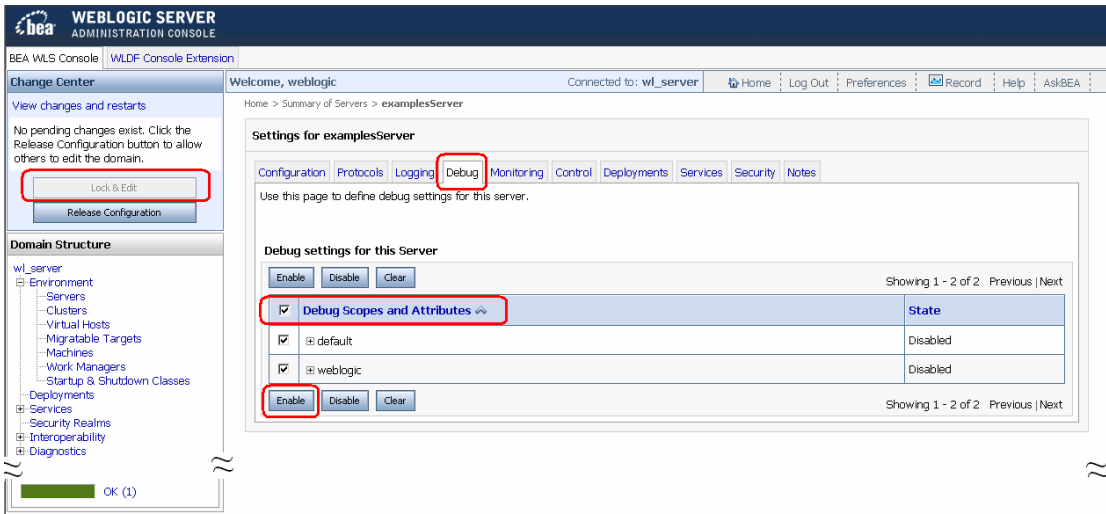
Step	Description
2.	<p>Click on the <b>Advisory Service</b> link in the <b>Communications Process Manager Client</b> window. Verify that the <b>Advisory Service</b> web page opens, and correctly displays. Fill in the fields with appropriate values. Verify that the recipients receive and acknowledge the advisory. A minimum of two users need to be created using the steps in Section 4, to perform these verifications.</p> 
3.	<p>Repeat Step 2 only clicking on the <b>Notify and Conference</b> link in the <b>Communications Process Manager Client</b> web page. Verify that the <b>Notify and Conference</b> service can be launched (not shown). Verify that the <b>Notify and Conference</b> service successfully completes.</p>
4.	<p>Repeat Step 2 only clicking on the <b>Notify and Respond</b> link in the <b>Communications Process Manager Client</b> web page. Verify that the <b>Notify and Respond</b> service can be launched (not shown). Verify that the <b>Notify and Respond</b> service is successfully completed.</p>
5.	<p>Repeat Step 2 only clicking on the <b>Find and Call</b> link in the <b>Communications Process Manager Client</b> web page. Verify that the <b>Find and Call</b> service can be launched (not shown). Verify that the <b>Find and Call</b> service is successfully completed.</p>

Step	Description																														
6.	<p>Launch a web browser and enter the URL <b>http://&lt; CPM server IP address &gt;/VIA</b> to open the CPM server login page. Enter the user credentials for the CPM user account used in the above verification steps. The <b>Communications Process Manager Inbox</b> web page opens. Click <b>Outbox</b> and then select <b>Completed</b> from the <b>Select</b> drop down list to view the completed notifications.</p> <p>The completed notifications are displayed as shown below. To view the contents of a notification, click the subject of the notification.</p> <div data-bbox="316 550 1417 1486" style="border: 1px solid black; padding: 10px;">  <p>The screenshot shows the Avaya Communications Process Manager Outbox interface. At the top, there is a navigation bar with 'Inbox', 'Outbox', 'Account', and 'Administration' on the left, and 'Help' and 'Logout' on the right. Below the navigation bar, there is a 'Select:' dropdown menu set to 'Completed' and a 'Help' link. A message states 'You have 4 completed requests.' Below this is a table with 6 columns: 'Application', 'Session Id', 'Start Date', 'End Date', and 'Subject'. Each row has a checkbox in the first column. The table contains 4 rows of data. Below the table is a 'Delete Selected Request(s)' button. At the bottom of the interface, there is a footer with 'Jenny' on the left and 'CPM v2.1' on the right, and a copyright notice 'Copyright © 2007 Avaya, Inc. All Rights Reserved.' in the center.</p> <table border="1" data-bbox="349 871 1388 1260"> <thead> <tr> <th><input type="checkbox"/></th> <th>Application</th> <th>Session Id</th> <th>Start Date</th> <th>End Date</th> <th>Subject</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/> 1</td> <td>Click To Find Service</td> <td>AAAAAKvvYpo=Hdbezw==</td> <td>Wednesday 05 March 2008 , 5:20:39 pm</td> <td>Wednesday 05 March 2008 , 5:21:14 pm</td> <td><a href="#">Invitation to conference call.</a></td> </tr> <tr> <td><input type="checkbox"/> 2</td> <td>Notification And Response Service</td> <td>AAAAAKvvYpk=HdbeUw==</td> <td>Wednesday 05 March 2008 , 5:18:50 pm</td> <td>Wednesday 05 March 2008 , 5:19:19 pm</td> <td><a href="#">Customer Survey</a></td> </tr> <tr> <td><input type="checkbox"/> 3</td> <td>Notify And Conference Service</td> <td>AAAAAKvvYpg=Hdbd1w==</td> <td>Wednesday 05 March 2008 , 5:14:56 pm</td> <td>Wednesday 05 March 2008 , 5:15:45 pm</td> <td><a href="#">Product Release Delayed</a></td> </tr> <tr> <td><input type="checkbox"/> 4</td> <td>Advisory Service</td> <td>AAAAAKvvYpc=HdbdWw==</td> <td>Wednesday 05 March 2008 , 5:12:41 pm</td> <td>Wednesday 05 March 2008 , 5:13:03 pm</td> <td><a href="#">Weather Advisory</a></td> </tr> </tbody> </table> </div>	<input type="checkbox"/>	Application	Session Id	Start Date	End Date	Subject	<input type="checkbox"/> 1	Click To Find Service	AAAAAKvvYpo=Hdbezw==	Wednesday 05 March 2008 , 5:20:39 pm	Wednesday 05 March 2008 , 5:21:14 pm	<a href="#">Invitation to conference call.</a>	<input type="checkbox"/> 2	Notification And Response Service	AAAAAKvvYpk=HdbeUw==	Wednesday 05 March 2008 , 5:18:50 pm	Wednesday 05 March 2008 , 5:19:19 pm	<a href="#">Customer Survey</a>	<input type="checkbox"/> 3	Notify And Conference Service	AAAAAKvvYpg=Hdbd1w==	Wednesday 05 March 2008 , 5:14:56 pm	Wednesday 05 March 2008 , 5:15:45 pm	<a href="#">Product Release Delayed</a>	<input type="checkbox"/> 4	Advisory Service	AAAAAKvvYpc=HdbdWw==	Wednesday 05 March 2008 , 5:12:41 pm	Wednesday 05 March 2008 , 5:13:03 pm	<a href="#">Weather Advisory</a>
<input type="checkbox"/>	Application	Session Id	Start Date	End Date	Subject																										
<input type="checkbox"/> 1	Click To Find Service	AAAAAKvvYpo=Hdbezw==	Wednesday 05 March 2008 , 5:20:39 pm	Wednesday 05 March 2008 , 5:21:14 pm	<a href="#">Invitation to conference call.</a>																										
<input type="checkbox"/> 2	Notification And Response Service	AAAAAKvvYpk=HdbeUw==	Wednesday 05 March 2008 , 5:18:50 pm	Wednesday 05 March 2008 , 5:19:19 pm	<a href="#">Customer Survey</a>																										
<input type="checkbox"/> 3	Notify And Conference Service	AAAAAKvvYpg=Hdbd1w==	Wednesday 05 March 2008 , 5:14:56 pm	Wednesday 05 March 2008 , 5:15:45 pm	<a href="#">Product Release Delayed</a>																										
<input type="checkbox"/> 4	Advisory Service	AAAAAKvvYpc=HdbdWw==	Wednesday 05 March 2008 , 5:12:41 pm	Wednesday 05 March 2008 , 5:13:03 pm	<a href="#">Weather Advisory</a>																										

Step	Description
7.	<p>The following log files are available on the Avaya CPM for troubleshooting purposes. The log files are stored under the <b>/var/log/cpm</b> directory.</p> <ul style="list-style-type: none"> <li>• mail.log</li> <li>• nafsvc.log</li> <li>• oam.log</li> <li>• dcore.log</li> <li>• commflow.log</li> <li>• advsvc.log</li> <li>• b2bua.log</li> <li>• account.log</li> <li>• via.log</li> <li>• platform-licensing.log</li> <li>• user-licensing.log</li> <li>• cs-tomcat-memorymonitor.log</li> <li>• cpm.log</li> <li>• complete.log</li> </ul>

Step	Description																														
8.	<p>BEA Weblogic Application Server's logging is available via the BEA Weblogic administration console web interface. To view BEA Weblogic logs select <b>Diagnostics</b> → <b>Log Files</b>. The <b>Summary of log files</b> opens in the right frame. Select the radio button for the log file of interest, and then click <b>View</b>.</p>  <p>The screenshot shows the BEA Weblogic Administration Console interface. The top navigation bar includes 'BEA WLS Console' and 'WLDLF Console Extension'. The left sidebar contains sections for 'Change Center', 'Domain Structure', 'How do I...', and 'System Status'. The 'Domain Structure' section shows a tree view where 'Log Files' is selected under 'Diagnostics'. The main content area displays the 'Summary of Log Files' page, which includes an introductory paragraph, a 'Customize this table' link, and a table of log files. The table has columns for 'Name', 'Type', and 'Server'. The 'DomainLog' entry is selected with a radio button.</p> <table border="1" data-bbox="665 877 1339 1249"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Server</th> </tr> </thead> <tbody> <tr> <td><input type="radio"/> DomainLog</td> <td>Domain Log</td> <td>examplesServer</td> </tr> <tr> <td><input type="radio"/> EventsDataArchive</td> <td>Instrumentation</td> <td>examplesServer</td> </tr> <tr> <td><input type="radio"/> HarvestedDataArchive</td> <td>Metric Data</td> <td>examplesServer</td> </tr> <tr> <td><input type="radio"/> HTTPAccessLog</td> <td>HTTP Access</td> <td>examplesServer</td> </tr> <tr> <td><input type="radio"/> JMSMessageLog/examplesJMServer</td> <td>JMS Log</td> <td>examplesServer</td> </tr> <tr> <td><input type="radio"/> JMSMessageLog/JMSServer1</td> <td>JMS Log</td> <td>examplesServer</td> </tr> <tr> <td><input type="radio"/> JMSMessageLog/WseeJMServer</td> <td>JMS Log</td> <td>examplesServer</td> </tr> <tr> <td><input type="radio"/> JMSSAFMessageLog/ReliableWseeSAFAgent</td> <td>JMS SAF Agent Log</td> <td>examplesServer</td> </tr> <tr> <td><input type="radio"/> ServerLog</td> <td>Server Log</td> <td>examplesServer</td> </tr> </tbody> </table>	Name	Type	Server	<input type="radio"/> DomainLog	Domain Log	examplesServer	<input type="radio"/> EventsDataArchive	Instrumentation	examplesServer	<input type="radio"/> HarvestedDataArchive	Metric Data	examplesServer	<input type="radio"/> HTTPAccessLog	HTTP Access	examplesServer	<input type="radio"/> JMSMessageLog/examplesJMServer	JMS Log	examplesServer	<input type="radio"/> JMSMessageLog/JMSServer1	JMS Log	examplesServer	<input type="radio"/> JMSMessageLog/WseeJMServer	JMS Log	examplesServer	<input type="radio"/> JMSSAFMessageLog/ReliableWseeSAFAgent	JMS SAF Agent Log	examplesServer	<input type="radio"/> ServerLog	Server Log	examplesServer
Name	Type	Server																													
<input type="radio"/> DomainLog	Domain Log	examplesServer																													
<input type="radio"/> EventsDataArchive	Instrumentation	examplesServer																													
<input type="radio"/> HarvestedDataArchive	Metric Data	examplesServer																													
<input type="radio"/> HTTPAccessLog	HTTP Access	examplesServer																													
<input type="radio"/> JMSMessageLog/examplesJMServer	JMS Log	examplesServer																													
<input type="radio"/> JMSMessageLog/JMSServer1	JMS Log	examplesServer																													
<input type="radio"/> JMSMessageLog/WseeJMServer	JMS Log	examplesServer																													
<input type="radio"/> JMSSAFMessageLog/ReliableWseeSAFAgent	JMS SAF Agent Log	examplesServer																													
<input type="radio"/> ServerLog	Server Log	examplesServer																													

Step	Description																																																							
9.	<p>The selected log file opens in the right frame for viewing. All entries in the log file can be searched by using the <b>Previous</b> and <b>Next</b> selections at the top of the log entries table. Select the radio button for the event of interest, and click <b>View</b> to view the event details (not shown).</p>  <p>The screenshot shows the BEA WebLogic Server Administration Console. The main content area displays the 'Server Log' for the 'examplesServer'. The log entries table is as follows:</p> <table border="1"> <thead> <tr> <th>Date</th> <th>Subsystem</th> <th>Severity</th> <th>Message ID</th> <th>Message</th> </tr> </thead> <tbody> <tr> <td>Feb 28, 2008 5:40:33 PM EST</td> <td>ServletContext- /clientsdk- webui</td> <td>Info</td> <td>BEA-000000</td> <td>JspServlet: param verbose initialized to: true</td> </tr> <tr> <td>Feb 28, 2008 5:40:33 PM EST</td> <td>ServletContext- /clientsdk- webui</td> <td>Info</td> <td>BEA-000000</td> <td>JspServlet: param packagePrefix initialized to: jsp_servlet</td> </tr> <tr> <td>Feb 28, 2008 5:40:33 PM EST</td> <td>ServletContext- /clientsdk- webui</td> <td>Info</td> <td>BEA-000000</td> <td>JspServlet: param compilerclass initialized to: null</td> </tr> <tr> <td>Feb 28, 2008 5:40:33 PM EST</td> <td>ServletContext- /clientsdk- webui</td> <td>Info</td> <td>BEA-000000</td> <td>JspServlet: param compileCommand initialized to: javac</td> </tr> <tr> <td>Feb 28, 2008 5:40:33 PM EST</td> <td>ServletContext- /clientsdk- webui</td> <td>Info</td> <td>BEA-000000</td> <td>JspServlet: param compilerlval initialized to: javac</td> </tr> <tr> <td>Feb 28, 2008 5:40:33 PM EST</td> <td>ServletContext- /clientsdk- webui</td> <td>Info</td> <td>BEA-000000</td> <td>JspServlet: param pageCheckSeconds initialized to: 1</td> </tr> <tr> <td>Feb 28, 2008 5:40:33 PM EST</td> <td>ServletContext- /clientsdk- webui</td> <td>Info</td> <td>BEA-000000</td> <td>JspServlet: param encoding initialized to: null</td> </tr> <tr> <td>Feb 28, 2008 5:40:33 PM EST</td> <td>ServletContext- /clientsdk- webui</td> <td>Info</td> <td>BEA-000000</td> <td>JspServlet: param superclass initialized to: null</td> </tr> <tr> <td>Feb 28, 2008 5:40:33 PM EST</td> <td>ServletContext- /clientsdk- webui</td> <td>Info</td> <td>BEA-000000</td> <td>JspServlet: param workingDir initialized to: C:\bea\wlserver_10.0\samples\domains\wl_server\servers\examplesServer\tmp\_WL_user\clientsdk- webui\mkdef8</td> </tr> <tr> <td>Feb 28, 2008 5:40:33 PM EST</td> <td>ServletContext- /clientsdk- webui</td> <td>Info</td> <td>BEA-000000</td> <td>JspServlet: initialization complete</td> </tr> </tbody> </table>	Date	Subsystem	Severity	Message ID	Message	Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param verbose initialized to: true	Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param packagePrefix initialized to: jsp_servlet	Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param compilerclass initialized to: null	Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param compileCommand initialized to: javac	Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param compilerlval initialized to: javac	Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param pageCheckSeconds initialized to: 1	Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param encoding initialized to: null	Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param superclass initialized to: null	Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param workingDir initialized to: C:\bea\wlserver_10.0\samples\domains\wl_server\servers\examplesServer\tmp\_WL_user\clientsdk- webui\mkdef8	Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: initialization complete
Date	Subsystem	Severity	Message ID	Message																																																				
Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param verbose initialized to: true																																																				
Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param packagePrefix initialized to: jsp_servlet																																																				
Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param compilerclass initialized to: null																																																				
Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param compileCommand initialized to: javac																																																				
Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param compilerlval initialized to: javac																																																				
Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param pageCheckSeconds initialized to: 1																																																				
Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param encoding initialized to: null																																																				
Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param superclass initialized to: null																																																				
Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: param workingDir initialized to: C:\bea\wlserver_10.0\samples\domains\wl_server\servers\examplesServer\tmp\_WL_user\clientsdk- webui\mkdef8																																																				
Feb 28, 2008 5:40:33 PM EST	ServletContext- /clientsdk- webui	Info	BEA-000000	JspServlet: initialization complete																																																				

Step	Description
10.	The <b>servername.log</b> , <b>domainname.log</b> , and <b>access.log</b> files also are available in the <b>.../domains/domainName/servers/serverName/logs</b> directory on the server. The BEA Weblogic administration console is not required to view these logs. The log files located in this directory can be opened for viewing using a text editor on the server.
11.	Debug messages can be written to the log files. In the <b>Domain Structure</b> frame select the <b>Environment</b> → <b>Servers</b> item to open the <b>Summary of Servers</b> frame on the right (not shown). In the <b>Name</b> column of the servers table click on the <b>server link</b> of interest (e.g., <b>examplesServer</b> ).
12.	<p>The <b>Settings for server</b> (e.g., <b>examplesServer</b>) frame opens on the right. At the top of the frame select the <b>Debug</b> tab to display the <b>Debug settings for the Server</b> table.</p> <p>In the <b>Change Center</b> frame click the <b>Lock &amp; Edit</b> button, to activate the fields on this page. Then check the <b>Debug Scopes and Attributes</b> of interest. Checking the checkbox in the title row of the table selects all the debug scopes. Click the <b>Enable</b> button to enable logging of debug messages.</p> 
13.	Click the <b>Activate Changes</b> button in the <b>Change Center</b> frame (not shown) to activate the logging of debug messages.
14.	Repeat the above steps only clicking <b>Disable</b> to disable the logging of debug messages.

## 10. Conclusion

These Application Notes show how to build and deploy the sample Avaya CPM Client web application on a BEA Weblogic Application Server, using BEA Workshop. In addition, it is shown that the Web Applications deployed on a BEA Weblogic Application server interoperate with Avaya Communications Process Manager via web services.

## 11. Additional References

The following Avaya product documentation is available at <http://support.avaya.com>.

- [1] “Administrator Guide for Avaya Communication Manager”, Issue 3.1, Doc ID: 03-300509, February 2007.
- [2] “Avaya Communications Process Manager Installation and Configuration Guide Release 2.1”, Issue 1, Oct 2007, Document Number 04-601158.
- [3] “Avaya Communications Process Manager Online help for the Administrative Web Portal Release 2.1”, Issue 4, Oct 2007, Document Number 04-601163.
- [4] “SIP Enablement Services (SES) Implementation Guide”, Issue 4, Doc ID : 16-300140, May 2007.
- [5] “Avaya Voice Portal 4.0 Documentation Library”, Jun-2007.
- [6] “Avaya Meeting Exchange Express Edition Release 1.5 Installation and Configuration Guide”, Issue 1, Doc ID: 04-601898, March 2007.
- [7] Avaya DevConnect web site <http://www.avaya.com/devconnect>.
- [8] “Avaya Communications Process Manager Release 2.1 Application Programmer’s Guide”, Issue 1, Doc ID: 04-602358, Oct 2007.

The following documents may be obtained from <http://e-docs.bea.com/wls/docs100/> and <http://e-docs.bea.com/wlw/docs100/>.

- [9] “BEA Products Installation Guide”, Version 10.0, March, 2007.
- [10] “BEA Weblogic Server: Introduction to BEA Weblogic Server and BEA Weblogic Express”, Version 10.0, March, 2007.
- [11] “BEA Weblogic Server: Deploying Applications to Weblogic Server”, Version 10.0, March, 2007.
- [12] “BEA Weblogic Server: Developing Applications with Weblogic Server”, Version 10.0, March, 2007.
- [13] “BEA Weblogic Server and Weblogic Express: Developing Web Applications, Serverlets, and JSPs for Weblogic Server”, Version 10.0, March, 2007.
- [14] “BEA Workshop User’s Guide”, Version 10.0, March, 2007 available at <http://e-docs.bea.com/wlw/docs100/guide/index.html>.

---

**©2008 Avaya Inc. All Rights Reserved.**

Avaya and the Avaya Logo are trademarks of Avaya Inc. All trademarks identified by ® and ™ are registered trademarks or trademarks, respectively, of Avaya Inc. All other trademarks are the property of their respective owners. The information provided in these Application Notes is subject to change without notice. The configurations, technical data, and recommendations provided in these Application Notes are believed to be accurate and dependable, but are presented without express or implied warranty. Users are responsible for their application of any products specified in these Application Notes.

Please e-mail any questions or comments pertaining to these Application Notes along with the full title name and filename, located in the lower right corner, directly to the Avaya Solution & Interoperability Test Lab at [interoplabnotes@list.avaya.com](mailto:interoplabnotes@list.avaya.com)