ORACLE®
INSURANCE

Oracle® Insurance Policy Administration

WebSphere Deployment

Installation Instructions – Step 2

Version 10.0.1.0

Documentation Part Number: E52368-01

February, 2014

ORACLE®

Table of Contents

INTRODUCTION

The Oracle Insurance Policy Administration (OIPA) application and the Oracle Insurance Rules Palette form a solution for configuring, managing and processing policy data. Both applications, along with the Web Application Utility, must be installed and then configured to work together.

This install guide will cover step two of the installation process, in which the OIPA application and Web Application Utility are configured using IBM WebSphere Version 8.5.5.0. Please refer to the OIPA Database Installation Instructions provided in the documentation library to ensure the database is set up correctly.

Customer Support

If you have any questions about the installation or use of our products, please visit the My Oracle Support website: https://support.oracle.com, or call (800) 223-1711.

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Prerequisites

Before beginning the installation steps, you must have the following components:

- A server with a Windows or Linux operating system
- WebSphere Application Server Version 8.5.5.0
- Administrative rights to the server
- Oracle Insurance Rules Palette V10.0.0.0 Media Pack from the Oracle Software Delivery Cloud. The Web Application Utility files are included in this Media Pack.
- Oracle Insurance Policy Administration V10.0.0.0 Media Pack from the Oracle Software Delivery Cloud

INITIAL SYSTEM CONFIGURATION

Database Drivers

Create a directory for the database drivers for your database. Copy the necessary driver .jar files into this directory.

Example: AIX or Linux: /opt/oracle/db_drivers

Windows: C:\oracle\oipa\db_drivers

- **Oracle 11.2 –** The necessary driver, **ojdbc-11.2.0.2.jar**, is included in the **libs** directory of the OIPA Media Pack.
- **Microsoft SQL Server 2008** – download the latest **jtds.jar** file.

  a. Download **jtds.jar** from the following site: http://sourceforge.net/projects/jtds/.
  b. Click the green **Download** box on the right side of the screen.   A pop-up window will open.
  c. Save the .zip file.
  d. Open the downloaded .zip file and extract the file **jtds-1.2.6.jar** from the root of the .zip file.
  e. Rename the file **jtds.jar**.

- **IBM DB2** – The two necessary .jar files (**db2jcc** and **db2jcc_license_cu)** are included with the purchase of the DB2 software. These files are not available for download. Contact your IT department if you need assistance locating these files.

Note: Each version of DB2 requires specific drivers. The files above correspond with DB2 10.1.

OIPA Setup

Note: For a WebSphere 8.5.5.0 deployment, the .jar files need to be installed only once. If OIPA Is reinstalled, the existing files may be re-used.

1. Copy the following files to the external jar file directory (e.g. /opt/IBM/WebSphere/AppServer/lib/ext):

- eclipselink-2.3.0.jar
- javax.persistence-2.0.3.jar

2. Edit amm.filter.properties from /opt/IBM/WebSphere/AppServer/properties

  a. Remove or comment out the lines listed below:

```
#Ignore-Scanning-Archives = ant.jar,                       \
#                           axis.jar,                       \
#                           CICS.jar,                       \
#                           db2java.zip,                    \
#                           db2jcc.jar,                     \
#                           db2jcc_javax.jar,               \
#                           dfjcics.jar,                    \
#                           ibmjndi.jar,                    \
```

```
#                              jaxrpc.jar,                    \
#                              jakarta.oro.jar,               \
#                              ldapbp.jar,                    \
#                              mailapi.jar,                   \
#                              msbase.jar,                    \
#                              mssqlserver.jar,               \
#                              saaj.jar,                      \
#                              wsdl4j.jar,                    \
#                              sqljdbc.jar,                   \
#                              struts.jar,                    \
#                              struts-scaffold.jar
```

    b.   Add the lines listed below:

3.   Ignore-Scanning-Archives : antisamy-1.4.3.jar, antlr-3.4.jar, aopalliance-1.0.jar, CICS.jar, ape-10.0.0.0.0.jar, aspectjrt-1.6.11.jar, aspectjweaver-1.6.11.jar, backport-util-concurrent-2.2.jar, batik-css-1.7.jar, batik-ext-1.7.jar, batik-util-1.7.jar, bcel-5.1.jar, bll-10.0.0.0.0.jar, bval-core-0.4.jar, bval-jsr303-0.4.jar,bsh-core-2.0b4.jar, cglib-nodep-2.2.2.jar, coherence.common-2.1.1.jar, coherence.patterns.processing-1.4.2.jar, commons-beanutils-core-1.8.3.jar, commons-codec-1.4.jar, commons-configuration-1.5.jar, commons-dbcp-1.4.jar, commons-dbutils-1.3.jar, commons-digester-1.8.jar, commons-fileupload-1.2.jar, commons-httpclient-3.1.jar, commons-jxpath-1.3.jar, commons-lang3-3.1.jar, commons-pool-1.5.6.jar, cycle.agent-10.0.0.0.0.jar, cycle.interface-10.0.0.0.0.jar, dal-10.0.0.0.0.jar, dcl-10.0.0.0.0.jar, dom4j-1.6.1.jar, el-ri-1.2.jar, esapi-2.0GA.jar, ext-10.0.0.0.0.jar, extensibility-10.0.0.0.0.jar, FastInfoset-1.2.2.jar, freemarker-2.3.16.jar, global.dal-10.0.0.0.0.jar, global.dcl-10.0.0.0.0.jar, global.processing-10.0.0.0.0.jar, global.utl-10.0.0.0.0.jar, icefaces-comps-ee-EE-1.8.2.GA_P04.jar, icefaces-ee-EE-1.8.2.GA_P04.jar, icefaces-facelets-ee-EE-1.8.2.GA_P04.jar, icu4j-4.6.jar, janino-2.5.16.jar, jaxen-1.1.3.jar, jep-2.4.0.jar, jibx-bind-1.2.2.jar, jibx-extras-1.2.2.jar, jibx-run-1.2.2.jar, jsf-api-1.2_15-b01-FCS.jar, jsf-impl-1.2_15-b01-FCS.jar, krysalis-jCharts-1.0.0-alpha-1.jar, log4j-1.2.16.jar, mail-1.4.jar, math-10.0.0.0.0.jar, model-10.0.0.0.0.jar, nekohtml-1.9.12.jar, pas.ape-10.0.0.0.0.jar, pas.bll-10.0.0.0.0.jar, pas.cycle-10.0.0.0.0.jar, pas.dal-10.0.0.0.0.jar, pas.dcl-10.0.0.0.0.jar, pas.helper-10.0.0.0.0.jar, pas.model-10.0.0.0.0.jar, pas.oipa-10.0.0.0.7.3.jar, pas.uip-10.0.0.0.0.jar, resource-10.0.0.0.0.jar, saxon-9.1.0.8.jar, saxon-dom-9.1.0.8.jar, slf4j-api-1.6.1.jar, slf4j-log4j12-1.6.1.jar, spring-aop-3.1.0.RELEASE.jar, spring-asm-3.1.0.RELEASE.jar, spring-beans-3.1.0.RELEASE.jar, spring-context-3.1.0.RELEASE.jar, spring-core-3.1.0.RELEASE.jar, spring-expression-3.1.0.RELEASE.jar, spring-instrument-3.1.0.RELEASE.jar, spring-jdbc-3.1.0.RELEASE.jar, spring-orm-3.1.0.RELEASE.jar, spring-tx-3.1.0.RELEASE.jar, spring-web-3.1.0.RELEASE.jar, sre.interface-10.0.0.0.0.jar, sre-10.0.0.0.0.jar, uip-10.0.0.0.0.jar, utl-10.0.0.0.0.jar, validation-api-1.0.0.GA.jar, web-10.0.0.0.0.jar, xml-apis-ext-1.3.04.jar

4.   Create a directory on the WebSphere server to store various configuration files for OIPA (e.g., /opt/oracle/oipa/ or C:\oracle\oipa\).

5.   Create sub-directories inside the directory from Step 4 called **conf** and **libs**.

Note:	Make a note of the path to the libs directory. It will be used later in the classpath for setting up the OIPA Shared Library.

6.   Copy the following files from the installation media into the **conf** sub-directory:

- coherence-cache-config.xml
- coherence-config.xml
- PAS.properties

7. Copy the following files from the installation media to the **libs** sub-directory:

   ▪ antlr-3.4.jar
   ▪ commons-collections-3.2.1.jar
   ▪ commons-logging-1.1.1.jar
   ▪ el-api-2.2.jar
   ▪ log4j-1.2.16.jar
   ▪ spring-instrument-3.1.0.RELEASE.jar

8. Download aspectj-1.6.11.jar from
   http://www.eclipse.org/downloads/download.php?file=/tools/aspectj/aspectj-1.6.11.jar

   ▪ Open aspectj-1.6.11.jar with an unzipping software and copy aspectjrt.jar and aspectjweaver.jar
     from the lib folder into the **libs** sub-directory.

Note:	Delete aspectjrt.jar from ./IBM/WebSphere/AppServer/lib

9. Unzip OIPA distribution and again unzip opss_standalone.zip and copy below jars to **libs** folder

   o .. \opss_standalone\modules\oracle.pki_11.1.1\ oraclepki.jar

   o .. \opss_standalone\modules\oracle.osdt_11.1.1 \osdt_cert.jar

   o .. \opss_standalone\modules\oracle.osdt_11.1.1 \osdt_core.jar

10. Unzip OIPA distribution and again unzip coherence.zip and copy ..\ coherence\lib\coherence.jar, ..\
    coherence\lib\coherence.common-2.1.1.jar, ..\ coherence\lib\coherence.pattern.processing-1.4.2.jar into
    libs folder.

11. Use a text editor to open the **PAS.properties** file that you just copied to the server.

    The PAS.properties file contains properties for Oracle, SQL Server, and DB2 database types, with the
    Oracle settings active by default. The inactive settings are commented out with a '#' character at the
    start of each line. To change a setting, remove the '#' from the required property setting, and insert it at
    the beginning of the setting you want to de-activate.

12. The properties setting must match the type of database being used. The two properties that are used
     to do this are:

   ▪ application.databaseType
   ▪ jpa.databasePlatform

Note:	Refer to the System Properties document in the Oracle Insurance Policy Administration E40981_01 Documentation Library on the OTN for a complete list of all properties and allowed values.

13. Identify the default locale in the PAS.properties file. The locale selected will determine the translation
    that is loaded in the database for OIPA when it launches. The default setting is English.

   ▪ application.defaultLocale

14. If using an Oracle or Microsoft SQL Server database, please skip this step. If using a DB2 database,
    you will need to modify the PAS.properties file to include configuration for case-insensitive searching:

- Modify the PAS.properties file to change the following line from the default setting of "false" to "true":

```
search.field.text.caseInsensitive=true
```

Web Application Utility Setup

1. Create a directory on the WebSphere server to store various configuration files for the Web Application Utility (i.e., /opt/oracle/paletteconfig/ or C:\oracle\paletteconfig\).

2. Create sub-directories inside the directory from Step 1 called **conf, libs** and **uploads**.

3. Copy the **PaletteWebApplication.properties** file into the **conf** sub-directory.

4. Use a text editor to open the PaletteWebApplication.properties file and edit the **download.dir** property to point to the **uploads** sub-directory created in Step 2.

CREATE AND CONFIGURE OIPA

Using a web browser, connect to the Administrative Console using the appropriate server name and 9060 as the default port (Ex: http://server_name:9060/admin).

Create the OIPA Application Server

1. Select **Servers>New server**.

2. Select server type **WebSphere application server**.

3. Enter **OIPA** for the name of the server.

4. Click **Next**.

5. Select the default server template.

6. Click **Next**.

7. Ensure **Generate Unique Http Ports** is checked.

8. Confirm the new server by clicking **Finish**.

9. After the application has been created, review the messages at the top of the Application servers screen to confirm that the new server has been created successfully.

10. Navigate to the OIPA server by clicking on its name within the list of application servers.

11. Click the **Session management** link in the **Container Settings** section.

12. Verify that **Enable Cookies** is checked. This is checked by default.

13. Click the **Enable Cookies** link and verify that 'Restrict **cookies to HTTPS sessions'** is unchecked. (This is unchecked by default.)

14. Click **OK**.

15. Click **OK** to save the configuration changes.

Note:	These instructions assume that a fresh installation of the application server was performed. All configuration settings must use the default settings unless otherwise noted. If the application server has been used for previous deployments, you may want to create a new application server specifically for the deployment of the OIPA environment.

Configure WebSphere Environment Variable for Database Driver

A variable must be used to define the location where the database driver is located. The name of the variable depends on the type of database.

- **Oracle**: ORACLE_JDBC_DRIVER_PATH
- **IBM DB2**: DB2UNIVERSAL_JDBC_DRIVER_PATH and DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH
- **Microsoft SQL Server:** User-defined_JDBC_DRIVER_PATH

1. Select **Environment>WebSphere Variables**.

2. Select the scope of the OIPA server from the drop-down list.

3. Click **New**.

4. Enter the name of the database driver variable, as listed above.

5. In the **Value** field, enter the path to the directory where the database driver is located. For example:

   `/opt/oracle/db_drivers` or `C:\oracle\oipa\db_drivers`

6. Click **OK**.

Configure the OIPA Application Server

JVM Settings

1. Navigate to **Servers>Server Types>WebSphere application servers**.

2. Select the OIPA application server.

3. Expand **Java and Process Management**.

4. Select **Process Definition**.

5. Select **Java Virtual Machine**.

6. In the **Classpath** text box, enter the location of the OIPA property files.

   Example: /opt/oracle/oipa/conf/ or c:\oracle\oipa\conf\

Note:	Be sure to include the trailing slash (/ or \) character.

7. Set **Initial Heap Size** to 512.

8. Set **Maximum Heap Size** to 2048.

9. In the **Generic JVM Arguments** text box, enter the following arguments, replacing the location of each file (highlighted) with the correct location for the configuration.

Note:	You may find it easier to copy the text below to a text editor and make the necessary changes there, then copy and paste it into the JVM Arguments text box.

```
-Duser.language=en -Duser.region=US -Djava.net.preferIPv4Stack=true
-Djava.net.preferPv6Addresses=false
-javaagent:/opt/oracle/oipa/libs/spring-instrument-3.1.0.RELEASE.jar
-Dtangosol.coherence.override=/opt/oracle/oipa/conf/coherence-config.xml
-Dtangosol.coherence.cacheconfig=/opt/oracle/oipa/conf/coherence-cache-config.xml
-Dtangosol.pof.config=com-adminserver-pas-web-pof-config.xml
```

10. Click **OK**.

Configure the Listening Port

1. Select **Servers>Server Types>WebSphere application servers**.

2. Select the OIPA server.

3. In the **Communications** section, select **Ports**.

4. Note the port listed for **WC_defaulthost.**

   a. If the port number needs to be changed, select the WC_defaulthost port.
   b. Modify the **Port** text box as needed.
   c. Click **OK**.

Configure the Virtual Host

1. Select **Environment > Virtual Hosts**.

2. Click **default_host**.

3. Click **Host Aliases**.

4. Click **New**.

5. Enter the port assigned for WC_defaulthost in the previous listening port section.

6. Click **OK**.

Create Data Sources

Add the JDBC Provider

1. Select **Resources>JDBC>JDBC providers**.

2. Select the scope of the server from the drop-down list (the drop-down list displays if the "**Show scope selection drop-down list…**" is checked).

3. Click **New**.

4. Select the **Database type**.

   - For Oracle, select **Oracle**.
   - For DB2, select **DB2**.
   - For Microsoft SQL Server, select **user-defined**.

5. Select the **Provider type**.

   - For Oracle, select **Oracle JDBC Driver**.
   - For DB2, select **DB2 Universal JDBC Driver Provider**.

6. Select **Implementation type**.

   - For Oracle, select **XA data source**.
   - For DB2, select **XA data source**.
   - For Microsoft SQL Server, enter **net.sourceforge.jtds.jdbcx.JtdsDataSource** for the **Implementation class name**.

7. Click **Next**.

8. Enter the variable name used for the database driver.

   - For Oracle, enter ${ORACLE_JDBC_DRIVER_PATH}
   - For DB2, click **Next**.
   - For Microsoft SQL Server, enter ${User-defined_JDBC_DRIVER_PATH}

9. Click **Finish**

> **Note:** For Oracle and Microsoft SQL Server, skip steps 10 and 11. For DB2, continue at step 10.

10. Click on the newly created JDBC provider.

11. Confirm that the class path field is set to correctly reference the two DB2 drivers. Refer to the examples below. Edit the field if needed.

   - ${DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc.jar
   - ${DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cu.jar
   - ${DB2UNIVERSAL_JDBC_DRIVER_PATH}/db2jcc_license_cisuz.jar

12. Click **OK**.

Create the Data Sources

Four data sources must be created:

   - ADMINSERVERDS
   - ADMINSERVERRESOURCEDS
   - ADMINSERVERSEARCHDS
   - ADMINSERVERREADONLYDS.

A readonly database user should be used for ADMINSERVERRESOURCEDS, ADMINSERVERSEARCHDS and ADMINSERVERREADONLYDS.Repeat the following steps for each data source, replacing **<DATASOURCE_NAME>** with the specific data source being configured.

1. Select **Resources>JDBC>Data sources.**

2. Click **New**.

3. For the **data source name**, enter <DATASOURCE_NAME>.

4. For the **JNDI name**, enter <DATASOURCE_NAME>.

5. Click **Next**.

6. Select the JDBC provider created in the last step.

7. Click **Next**.

8. Enter the database connection information.

   - For Oracle, enter the database information in the following format:

     jdbc:oracle:thin:@**hostname:port:SID**

   - For DB2, enter the **database name, hostname** and **port**.
   - For Microsoft SQL Server, skip to the next step.

9. Uncheck the checkbox for **Use this data source in container managed persistence**.

10. Click **Next**.

11. On the **Setup security aliases** screen, click **Next**.

12. Click **Finish**.

13. Select <DATASOURCE_NAME> from the list of data sources that appears.

14. Select **Custom Properties**.

15. Click **New** to create new variables for each property listed below.

- For Oracle:

  o Name: **user**
    Value: database user name
  o Name: **password**
    Value: database password

- For DB2:

  o Name: **user**
    Value: database user name
  o Name: **password**
    Value: database password
  o Name: **currentSchema**
    Value: schema name (only needed for readonly data source.)

- For Microsoft SQL Server:

  o Name: **user**
    Value: database user name
  o Name: **password**
    Value: database password
  o Name: **databasename**
    Value: name of database
  o Name: **servername**
    Value: hostname of database server

16. In the Messages area at the top of the screen, click the link for **Save to the master configuration**.

17. Once complete, return to the **Datasources** screen and use the **Test Connection** button on the main data source list to confirm that the settings were successful for each data source.

Deploy the OIPA Application

1. Select **Applications>New Application**.

2. Select **New Enterprise Application**.

3. Use the **Local file system** or **Remote file system** browsing feature to locate the **PASJava-websphere.war** file and rename to PASJava.war.

4. Click **Next**.

5. Confirm that **Fast Path** is selected and click **Next**.
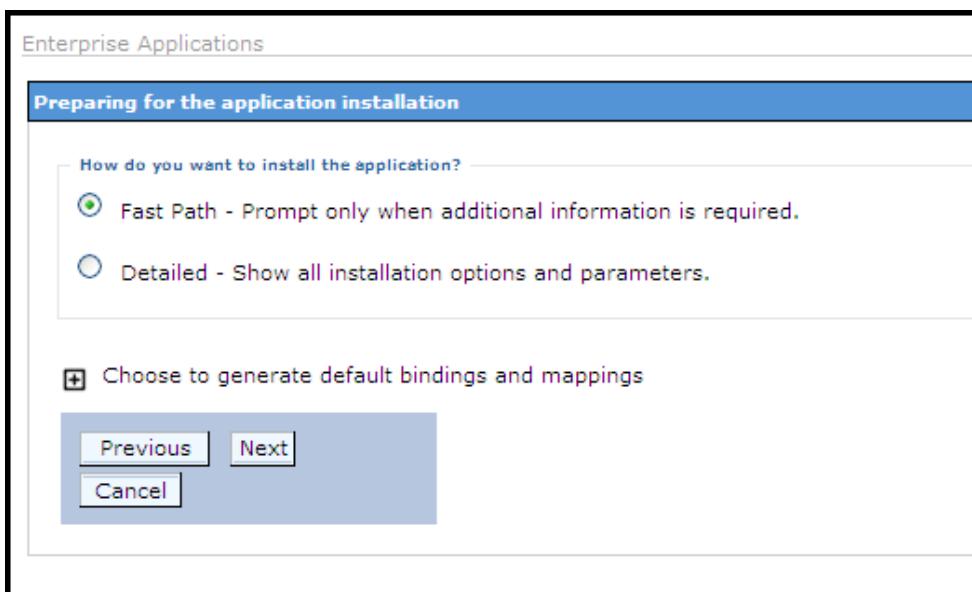
6. On the **Select installation options** screen, click **Next**.

7. On the Map modules to servers screen:

   - Select the server from the scrolling list.

   - Check the box for PASJava.war and click Apply.
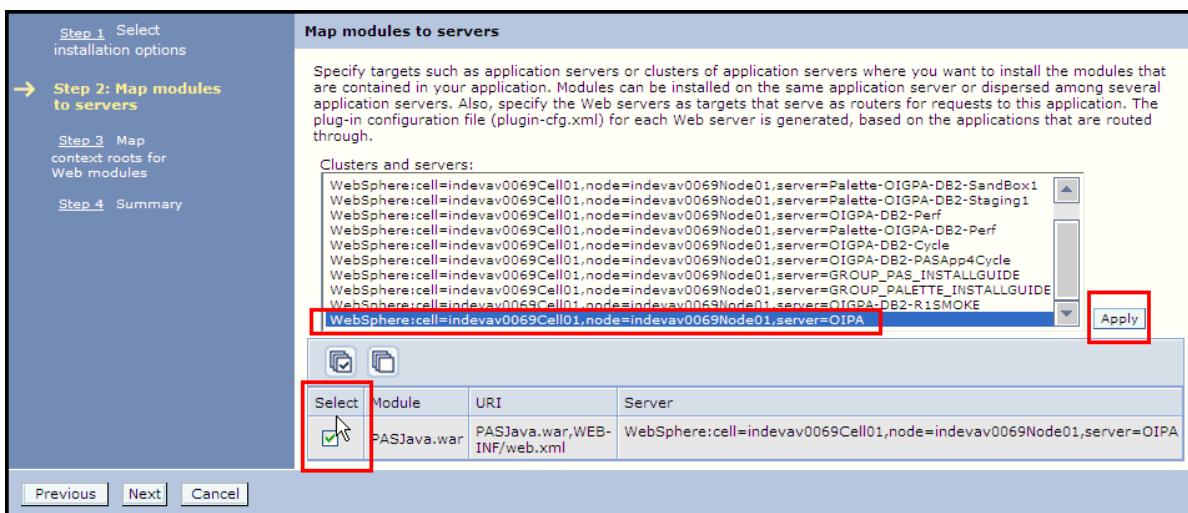
   - Click Next.

8. On the **Map resource references to resources** screen, enter proper datasources and click **Next**

9. On the **Map virtual hosts for Web modules** screen, check the box for PASJava.war and click **Next**.

10. On the **Map context roots for Web modules screen,** enter **PASJava** for the Context Root and click **Next**.
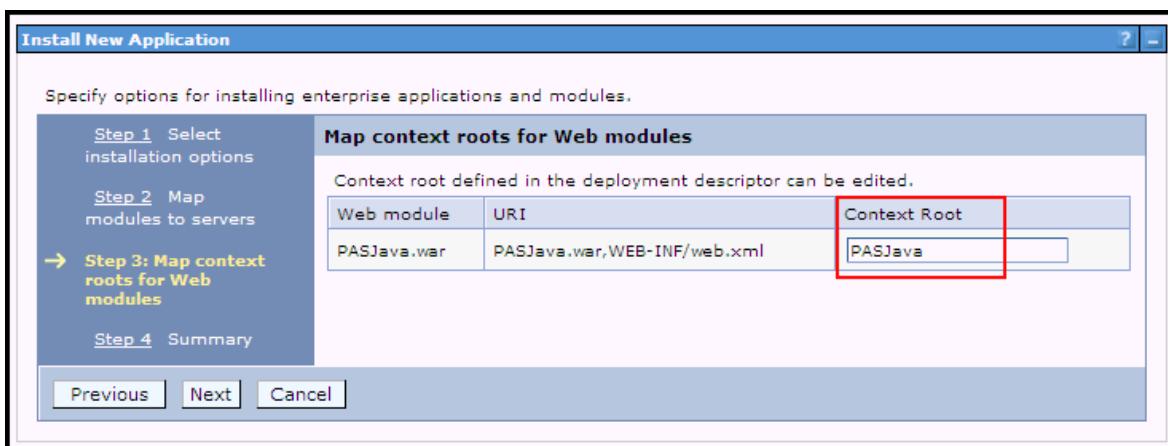
11. On the **Metadata for modules** screen, select checkbox of the metadata-complete attribute and click on **Next**

12. Confirm the settings on the Summary screen and click Finish.

Specify options for installing enterprise applications and modules.

Summary

Step 1 Select installation options	Summary of installation options
Step 2 Map modules to servers	
Step 3 Map resource references to resources	
Step 4 Map virtual hosts for Web modules	
Step 5 Map context roots for Web modules	
Step 6 Metadata for modules	
→ **Step 7: Summary**	

Options	Values
Precompile JavaServer Pages files	No
Directory to install application	
Distribute application	Yes
Use Binary Configuration	No
Deploy enterprise beans	No
Application name	PASJava-websphere_war1
Application edition	
Edition description	
Create MBeans for resources	Yes
Override class reloading settings for Web and EJB modules	No
Reload interval in seconds	
Deploy Web services	No
Validate Input off/warn/fail	warn
Process embedded configuration	No
File Permission	.*\.dll=755#.*\.so=755#.*\.a=755#.*\.sl=755
Application Build ID	Unknown
Allow dispatching includes to remote resources	No
Allow servicing includes from remote resources	No
Business level application name	
Asynchronous Request Dispatch Type	Disabled
Allow EJB reference targets to resolve automatically	No
Deploy client modules	No
Client deployment mode	Isolated
Validate schema	No
Cell/Node/Server	Click here

Previous | Finish | Cancel

13. The system lists output from the installation, with the final status at the end. Look for "Application PASJava_war installed successfully."

Installing...

If there are enterprise beans in the application, the EJB deployment process can take several minutes. Do not save the configuration until the process completes.

Check the SystemOut.log on the deployment manager or server where the application is deployed for specific information about the EJB deployment process as it occurs.
. . .

Application PASJava_war installed successfully.

To start the application, first save changes to the master configuration.

Changes have been made to your local configuration. You can:
* Save directly to the master configuration.
* Review changes before saving or discarding.

14. Click the link to **Save** directly to the master configuration.

Create a Shared Library for OIPA

1. Select **Environment > Shared Libraries**.

2. Select the OIPA application server from the selection box.

3. Click **New**.

4. For the **Name** field, enter **SharedLibs**.

5. For the **Classpath** field, enter the full paths to the following jar files (using the directory that was created in steps 4 and 5 in the OIPA Setup section):

    - antlr-3.4.jar
    - aspectjrt-1.6.11.jar
    - aspectjweaver-1.6.11.jar
    - commons-collections-3.2.1.jar
    - commons-logging-1.1.1.jar
    - log4j-1.2.16.jar
    - spring-instrument-3.1.0.RELEASE.jar
    - coherence.jar
    - coherence.common-2.1.1.jar
    - coherence.pattern.processing-1.4.2.jar
    - oraclepki.jar
    - osdt_cert.jar
    - osdt_core.jar

6. For the Native Library Path field, enter the full path for the following jar file:

    - el-api-2.2.jar

7. Click **OK**.

8. Save the configuration changes.

9. Select **Applications>WebSphere enterprise applications**.

10. Click on the OIPA application.

11. Under the References heading, click the **Shared library references** link.

12. Check the **PASJava.war** module and click the **Reference shared libraries** button.

13. In the Available box, click on **SharedLibs** and then click the **>>** button. This will move **SharedLibs** to the **Selected** box.

14. Click **OK**.

15. Click **OK**.

16. Save the configuration changes.

Configure the OIPA Application

1. Select **Applications>Application Types>WebSphere enterprise application**.

2. Select the **PASJava.war** application.

3. Click **Class loading and update detection.**

4. Check the radio button for **Classes loaded with application class loader first (parent last)**

5. Check the radio button for **Single class loader for application**.

6. Click **OK**.

7. Save the configuration.

Start the OIPA Application

1. Select **Servers>Server Types>WebSphere application servers**.

2. Select the OIPA application.

3. Click **Start**.

4. Look for the message: "<server name> server started successfully."

CREATE AND CONFIGURE THE WEB APPLICATION UTILITY

Using a web browser, connect to the Administrative Console using the appropriate server_name and port. (Ex: http://server_name:port/admin)

Note:	The files that are used to set up the Web Application Utility are often named PaletteConfig. Any files with the name PaletteConfig are part of the Web Application Utility.

Create the Web Application Utility Server

1. Select **Servers>New server**.

2. Select **WebSphere application server** for the server type and click **Next**.

3. Enter **PaletteConfig** for the name of the server.

4. Click **Next**.

5. Select the **default** server template.

6. Click **Next**.

7. Ensure **Generate Unique Ports** is checked.

8. Click **Next**.

9. Confirm the server settings and click **Finish**.

The system displays the **Application servers** screen.

Configure the Web Application Utility Server

JVM Settings

1. Select the PaletteConfig server.

2. Expand **Java and Process Management.**

3. Select **Process Definition**.

4. Select **Java Virtual Machine**.

5. In the **Classpath** text box, enter the location of the Web Application Utility property files.

6. Example: /opt/oracle/paletteconfig/conf/ or C:\oracle\paletteconfig\conf\

Note:	Be sure to include the trailing slash (/ or \) character.

7. Set **Initial Heap Size** to **256**.

8. Set **Maximum Heap Size** to **512**.

9. Click **OK**.

Configure the Listening Port

1. Select **Servers>Server Types>WebSphere application servers**.

2. Select the PaletteConfig server.

3. In the **Communications** section, select **Ports**.

4. Note, or change if needed, the port listed for **WC_defaulthost**.

5. Select **Finish**.

Configure the Virtual Host

1. Select **Environment>Virtual Hosts**.

2. Click **default_host**.

3. Click **Host Aliases**.

4. Click **New**.

5. Enter the port selected for **WC_defaulthost** in the previous step.

6. Click **OK**.

Deploy the Web Application Utility

1. Select **Applications>New Application**.

2. Select **New Enterprise Application**.

3. Use the Local file system and Remote file system browsing feature to locate and select the
   `PaletteConfig-waswl.war and rename to PaletteConfig.war` file.

4. Click **Next**.

5. Confirm that **Fast Path** is selected, and click **Next**.

6. On the **Select installation options** screen, click **Next** to accept the default options.

7. On the **Map modules to servers** screen:

   ▪ Select the server from the scrolling list.
   ▪ Check the box for PaletteConfig.war and click **Apply**
   ▪ Click **Next**.

8. On the **Map resource references to resources** screen, click next.

9. On the **Map virtual hosts for Web modules** screen, check the box for PaletteConfig-waswl.war and
   click **Next**.

10. On the **Map context roots for Web modules** screen, enter **PaletteConfig** for the Context Root and
    click **Next**.

11. **On the Metadata for modules screen**, select checkbox of the metadata-complete attribute and click **Next**

12. Confirm the settings on the Summary screen and Click Finish

13. The system lists output from the installation, with the final status at the end. Look for "Application PaletteConfig.war installed successfully."

14. Click the link to **Save** directly to the master configuration.
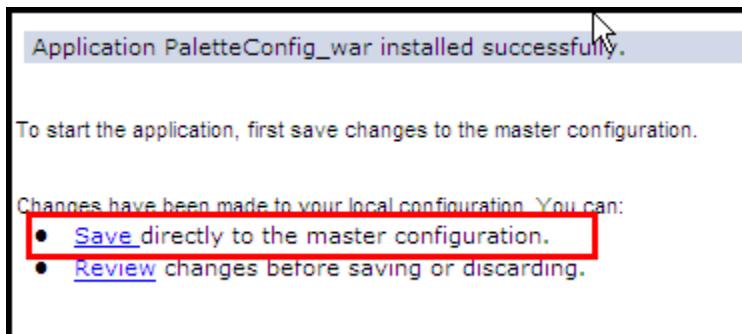
Create a Shared Library for Web Application Utility

1. Select **Environment>Shared Libraries**.

2. Select the Web Application Utility application server from the selection box.

3. Click **New**.

4. For the **Name** field, enter **SharedLibs**.

5. For the **Classpath** field, enter the full paths to the database driver jar files (using the directory that was created in the **Database Drivers** section on page 5). Be sure to supply the appropriate drivers for the database type you are using.

   ▪ AIX or Linux example for Oracle:

     `/opt/oracle/db_drivers/ojdbc-11.2.0.2.jar`

   ▪ Windows example for Oracle:

     `C:\oracle\db_drivers\ojdbc-11.2.0.2.jar`

6. Click **OK**.

7. Save the configuration changes.

8. Select **Applications>WebSphere enterprise applications**.

9. Click on the Web Application Utility application.

10. Under the **References** heading, click the **Shared library references** link.

11. Check the **PaletteConfig.war** module and click the **Reference shared libraries** button.

12. In the **Available** box, click on **SharedLibs** and then click the **>>** button. This will move SharedLibs to the **Selected** box.

13. Click **OK**.

14. Click **OK**.

15. Save the configuration changes.

Configure the Web Application Utility

1. If not already on this screen, select **Applications>Application Types>WebSphere enterprise applications**.

2. Select the PaletteConfig application.

3. Click **Class loading and update detection.**

4. Check the radio button for **Classes loaded with application class loader first (parent last).**

5. Check the radio button for **Single class loader for application**.

6. Click **OK**.

Start the Web Application Utility

1. Select **Servers>Application Types>WebSphere application servers**.

2. Select the PaletteConfig application.

3. Click **Start**.

4. Look for the message: "<server name> server started successfully."

TEST THE DEPLOYMENTS

The initial test of the deployment is to confirm that the application presents a login screen when the application URL is opened. Final testing of the OIPA deployment must wait until after the Web Application Utility is configured and initial users have been set up.

OIPA Deployment

1. Open a new Internet Explorer window.

2. Navigate to http://hostname:port/PASJava/.

Note:	If you change the context name, use that name in the URL instead of PASJava.

3. Confirm that the OIPA login screen appears.

Web Application Utility Deployment

1. Open a new Internet Explorer window.

2. Navigate to http://hostname:port/PaletteConfig/.

Note:	If you change the context name, use that name in the URL instead of PaletteConfig.

3. Confirm that the Web Application Utility set-up screen appears.

Important:	When configuring the Rules Palette environment use the host name and port number of the PaletteConfig server. The Web Application Utility must always be running in order for users to access the Rules Palette.