Legal Notices

Warranty Disclaimer

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

Restricted Rights Notice

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Hazardous Applications Notice

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Third Party Content, Products, and Services Disclaimer

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

.

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
- Administrative rights to the server
- Oracle Insurance Rules Palette v11.0.0.0 Media Pack from the Oracle Software Delivery Cloud. The Web Application Utility files are included in this Media Pack.
- Oracle Insurance Policy Administration v11.0.0.0 Media Pack from the Oracle Software Delivery Cloud

Note: For the Technology Stack details, please refer to the 11.0.0.0 version (part number E78460-01) information published in the Technology Stack section of the Oracle Technology Network (OTN).

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 12.1** – The necessary driver, **ojdbc7-12.1.0.1.jar**, is included in the libs directory of the OIPA Media Pack.
- **IBM DB2** – The two necessary .jar files (**db2jcc and db2jcc_license_cu**) are included with the purchase of the DB2 software.

> **Note:** These files are not available for download. Contact your IT department if you need assistance locating these files. Each version of DB2 requires specific drivers. The files above correspond with DB2 10.5.

OIPA Setup

Note: For a WebSphere 8.5.5.9 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-2.6.4.jar
   - javax.persistence-2.1.0.jar
2. Edit amm.filter.properties from /opt/IBM/WebSphere/AppServer/properties
3. 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
4. Add 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, mail-
api.jar, msbase.jar, mssqlserver.jar, saaj.jar, wsdl4j.jar, sqljdbc.jar, struts.jar,
struts-scaffold.jar, antisamy-1.5.3.jar, aopalliance-1.0.jar, ape-11.0.0.0.jar, back-
port-util-concurrent-2.2.jar, batik-css-1.8.jar, batik-ext-1.8.jar, batik-util-1.8.jar,
bcel-5.1.jar, bll-11.0.0.0.jar, bval-core-1.1.1.jar, bval-jsr303-0.5.jar, cglib-nodep-
3.2.4.jar, coherence-common-12.3.1.jar, coherence-processingpattern-12.3.1.jar, commons-
beanutils-core-1.8.3.jar, commons-codec-1.4.jar, commons-configuration-1.10.jar,
commons-dbcp-1.4.jar, commons-dbutils-1.6.jar, commons-digester-1.8.jar, commons-fileupload-1.6.jar, commons-httpclient-3.1.jar, commons-jxpath-1.3.jar, commons-lang3-3.2.jar, commons-pool-1.6.jar, cycle.agent-11.0.0.0.jar, cycle.interface-11.0.0.0.jar, dal-11.0.0.0.jar, dcl-11.0.0.0.jar, dom4j-1.6.1.jar, el-ri-1.2.jar, esapi-2.1.0.1.jar, ext-11.0.0.0.jar, extensibility-11.0.0.0.jar, FastInfoset-1.2.2.jar, freemarker-2.3.23.jar, global.dal-11.0.0.0.jar, global.dcl-11.0.0.0.jar, global.processing-11.0.0.0.jar, global.utl-11.0.0.0.jar, icu4j-4.6.jar, janino-2.5.16.jar, jaxb-api-2.0.jar, jaxen-1.1.3.jar, jaxws-api-2.1.jar, jep-2.4.0.jar, jibx-bind-1.2.6.jar, jibx-extras-1.2.6.jar, jibx-run-1.2.5.jar, krysalis-jCharts-1.0.0-alpha-1.jar, list.txt, log4j-1.2.17.jar, mail-1.4.jar, math-11.0.0.0.jar, model-11.0.0.0.jar, nekohtml-1.9.12.jar, pas.ape-11.0.0.0.jar, pas.bll-11.0.0.0.jar, pas.cycle-11.0.0.0.jar, pas.dal-11.0.0.0.jar, pas.dcl-11.0.0.0.jar, pas.di.ext-11.0.0.0.jar, pas.helper-11.0.0.0.jar, pas.model-11.0.0.0.jar, pas.oipa-11.0.0.0.jar, pas.uip-11.0.0.0.jar, resource-11.0.0.0.jar, saxon-9.1.0.8.jar, saxon-dom-9.1.0.8.jar, slf4j-api-1.7.6.jar, slf4j-log4j12-1.6.1.jar, spring-aop-4.2.3.RELEASE.jar, spring-asm-4.2.3.RELEASE.jar, spring-beans-4.2.3.RELEASE.jar, spring-context-4.2.3.RELEASE.jar, spring-context-support-4.2.3.RELEASE.jar, spring-core-4.2.3.RELEASE.jar, spring-expression-4.2.3.RELEASE.jar, spring-jdbc-4.2.3.RELEASE.jar, spring-orm-4.2.3.RELEASE.jar, spring-tx-4.2.3.RELEASE.jar, spring-web-4.2.3.RELEASE.jar, sre.interface-11.0.0.0.jar, sre-11.0.0.0.jar, uip-11.0.0.0.jar, utl-11.0.0.0.jar, validation-api-1.1.0.Final.jar, web-11.0.0.0.jar, xml-apis-ext-1.3.04.jar, xws-security-3.0.jar, javax.ws.rs-api-2.0.1.jar
```

5. Create a directory on the WebSphere server to store various configuration files for OIPA (e.g., /opt/oracle/oipa/ or C:\oracle\oipa).
6. 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.

7. Copy the following files from the installation media into the conf sub-directory:
   - coherence-cache-config.xml
   - coherence-config.xml
   - PAS.properties
   - logging.properties
8. Copy the following files from the installation media to the libs sub-directory:
   - antlr-3.5.2.jar
   - commons-collections-3.2.2.jar
   - commons-logging-1.2.jar
   - el-api-2.2.jar
   - log4j-1.2.17.jar
   - spring-instrument-4.2.3.RELEASE.jar
   - javax.ws.rs-api-2.0.1.jar
9. Download **aspectj-1.8.7.jar** from http://www.eclipse.org/downloads/download.php?file=/tools/aspectj/aspectj-1.8.7.jar

- Open aspectj-1.8.7.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

10. Unzip OIPA distribution and again unzip opss_standalone.zip and copy below jars to **libs** folder
    - .. \opss_standalone\modules\oracle.pki_11.1.1\ oraclepki.jar
    - .. \opss_standalone\modules\oracle.osdt_11.1.1 \osdt_cert.jar
    - .. \opss_standalone\modules\oracle.osdt_11.1.1 \osdt_core.jar
11. Unzip OIPA distribution and download Coherence.jar of version 12.2.1.1.0 from http://www.oracle.com/technetwork/middleware/coherence/downloads/coherence-archive-165749.html and copy ..\ coherence\lib\coherence.jar into **libs** folder.
12. 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, 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.
13. 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 E62439-01 Documentation Library on the OTN for a complete list of all properties and allowed values.
14. 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
15. If using an Oracle 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
16. Use a text editor to open the **logging.properties** file that you just copied to the server.
    - Edit the path mention for property "java.util.logging.FileHandler.pattern" with new application log location i.e. if application log location is /home/OIPA/logs then set this value as java.util.logging.FileHandler.pattern = %h/OIPA/logs/oipa%u.log. Also same time make sure that directory is created and has r/w permission.

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. Create a file PaletteWebApplication.properties in the conf subdirectory.

4. Open the PaletteWebApplication.properties file and add/edit the download.dir property to point to the uploads subdirectory created in Step 2.

Create and Configure OIPA

Create and Configure 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

1. 1.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
2. Select **Environment>WebSphere Variables**.
3. Select the scope of the OIPA server from the drop-down list.
4. Click **New**.
5. Enter the name of the database driver variable, as listed above.
6. 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

7. 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-4.2.3.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 -Dtangosol.coherence.mode=prod-Djava.util.logging.config.file="/opt/oracle/oipa/conf/logging.properties"

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**.
5. If the port number needs to be changed, select the WC_defaulthost port.
6. Modify the **Port** text box as needed.
7. 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**.
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**.
7. Click **Next**.
8. Enter the variable name used for the database driver.
   - For Oracle, enter ${ORACLE_JDBC_DRIVER_PATH}
   - For DB2, click **Next**.
9. Click Finish

> **Note:** For Oracle, 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

The following 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**.
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**:

      Name: user; Value: database user name

      Name: password; Value: database password
    - **For DB2**:

      Name: user; Value: database user name

      Name: password; Value: database password

      Name: currentSchema; Value: schema name (only needed for readonly data source)
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.

JMS Setup for Data Intake - Websphere Configuration

1. Introduction and Definitions

A JMS queue in Websphere Server is associated with a number of additional resources:

The high-level configuration steps are as follows to configure messaging within WebSphere Application Server for the MDB sample:

1. Configure the service bus
2. Configure the bus members
3. Configure the destinations
4. Verify the messaging engine startup
5. Configure JMS connection queue factory
6. Configure the destination JMS queue

A service integration bus supports applications using message-based and service-oriented architectures. A bus is a group of one or more interconnected servers or server clusters that are members of the bus. Applications connect to a bus at one of the messaging engines associated with its bus members.

Object Name	Type	JNDI Name
DataIntakeJMSServer	SIB (BUS Name)	
IntakeConnectionFactory	Queue Connection Factories	IntakeConnectionFactory
DIQueue	JMS Queue	DIQueue

2. Configuration Steps

Pre-requisites for running Data Intake:

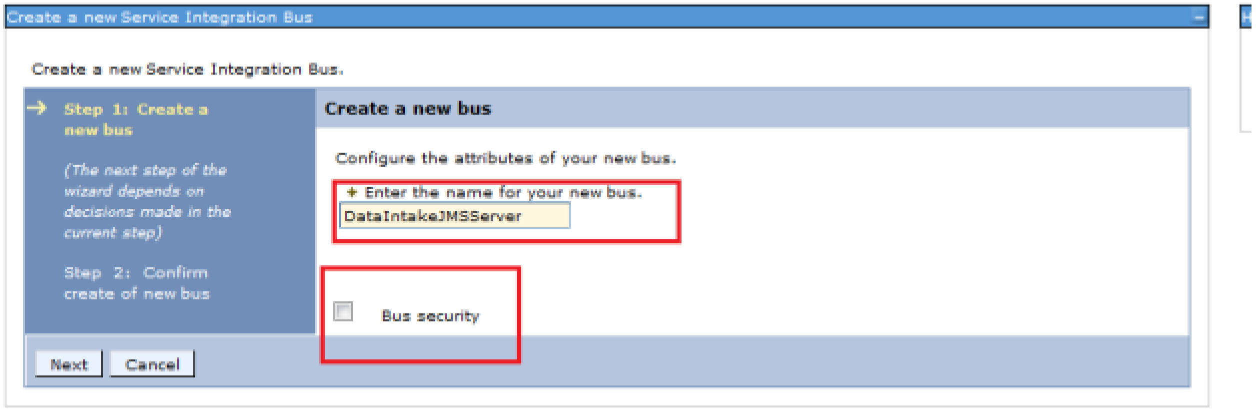
- SIB will be pointed to a Cycle server which should be up and running.
- The property dataIntake.enabled should be set to true on server side of Cycle.
- Need to provide Customer Name, Intake Profile name, and Class Name (In case of Enrollment) in the loading file, which should be created on OIPA side prior to Data Intake. The steps to be followed on OIPA side can be found in the below Page. Data Intake Guide

The following steps are done in the Websphere Application Console, beginning with the left-hand navigation menu.

Configure the service bus

1. From the **WebSphere Administrative Console**, select **Service integration > Buses**.
2. Click **New**.

3. Enter the following and then click **OK**:
   - **Name**: DataIntakeJMSServer
   - Uncheck **Bus Security**


Create a new Service Integration Bus

Create a new Service Integration Bus.

Step 1: Create a new bus

(The next step of the wizard depends on decisions made in the current step)

Step 2: Confirm create of new bus

Create a new bus

Configure the attributes of your new bus.

* Enter the name for your new bus.
DataIntakeJMSServer

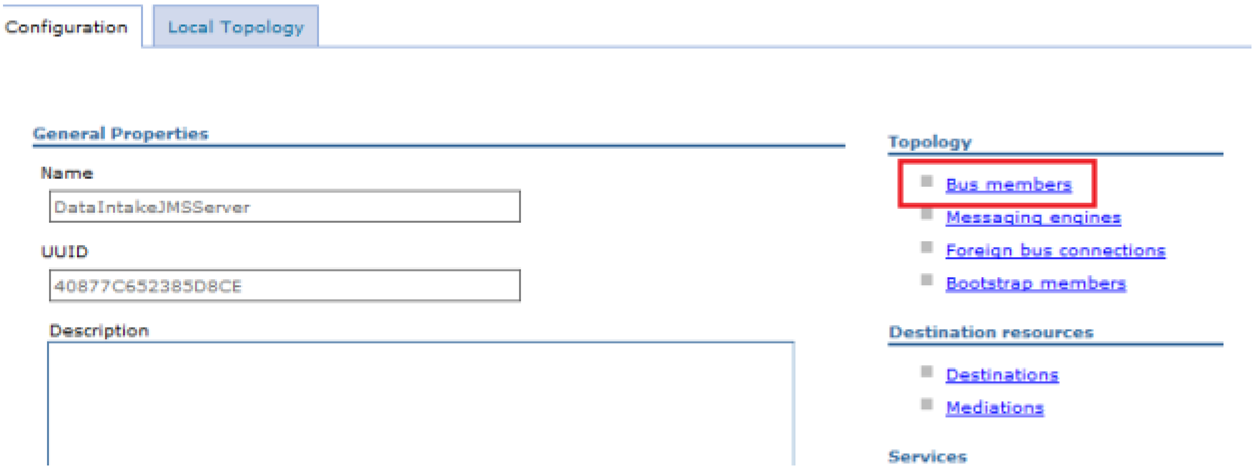
□ Bus security

4. Click on Next and Click on Finish on the next Screen. On successful creation. You can see the Bus as shown below:

Configure the bus members

To configure the Bus members, do the following:

1. Select **Service integration > Buses**.
2. Click the DataIntakeJMSServer created in the previous section.
3. Under **Topology** on right hand side click on Bus members



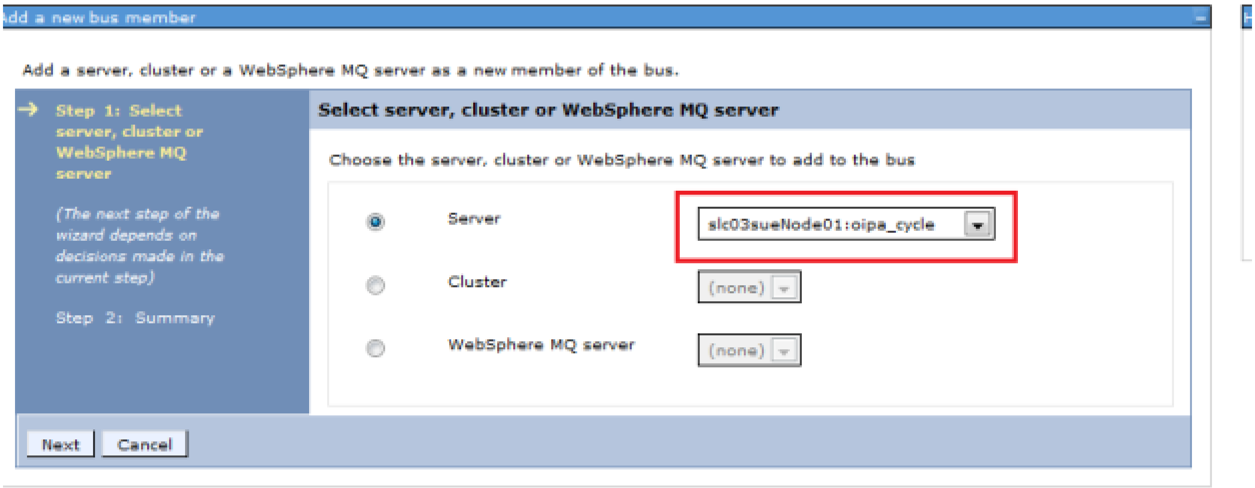
General Properties

Name
DataIntakeJMSServer

UUID
40877C652385D8CE

Description

4. On the next page select the cycle server for Server:



Add a new bus member

Add a server, cluster or a WebSphere MQ server as a new member of the bus.

→ **Step 1: Select server, cluster or WebSphere MQ server**

(The next step of the wizard depends on decisions made in the current step)

Step 2: Summary

Select server, cluster or WebSphere MQ server

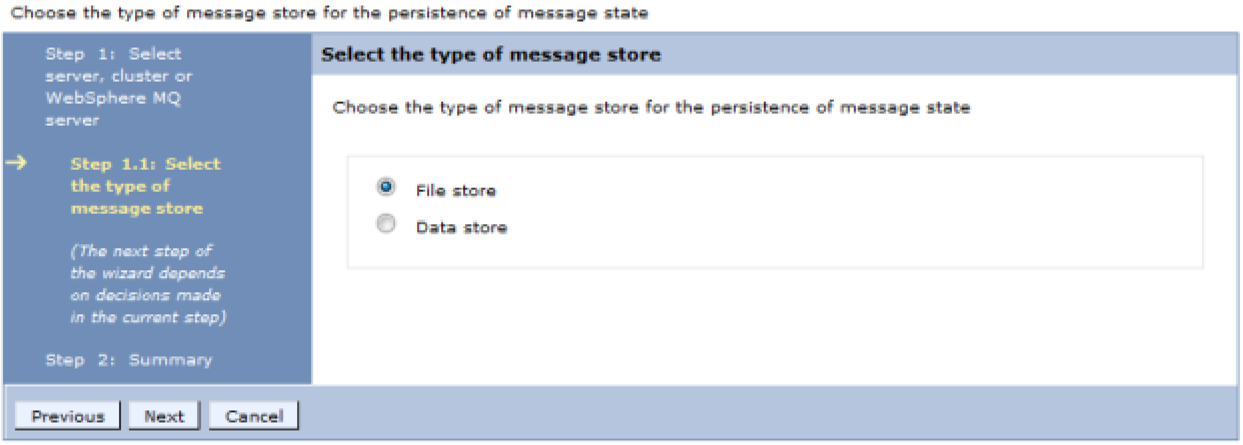
Choose the server, cluster or WebSphere MQ server to add to the bus

● Server: slc03sueNode01:oipa_cycle

○ Cluster: (none)

○ WebSphere MQ server: (none)

5. Click Next and in the next window select File Store



Choose the type of message store for the persistence of message state

Step 1: Select server, cluster or WebSphere MQ server

→ **Step 1.1: Select the type of message store**

(The next step of the wizard depends on decisions made in the current step)

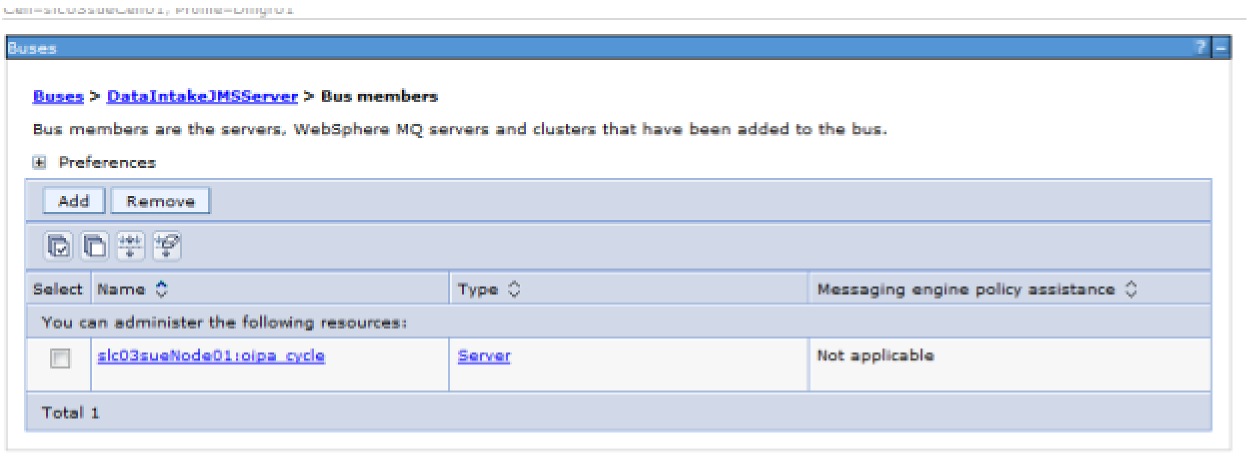
Step 2: Summary

Select the type of message store

Choose the type of message store for the persistence of message state

● File store

○ Data store



Cell=slc03sueCell01, Profile=Dmgr01

Buses

Buses > DataIntakeJMSServer > **Bus members**

Bus members are the servers, WebSphere MQ servers and clusters that have been added to the bus.

⊞ Preferences

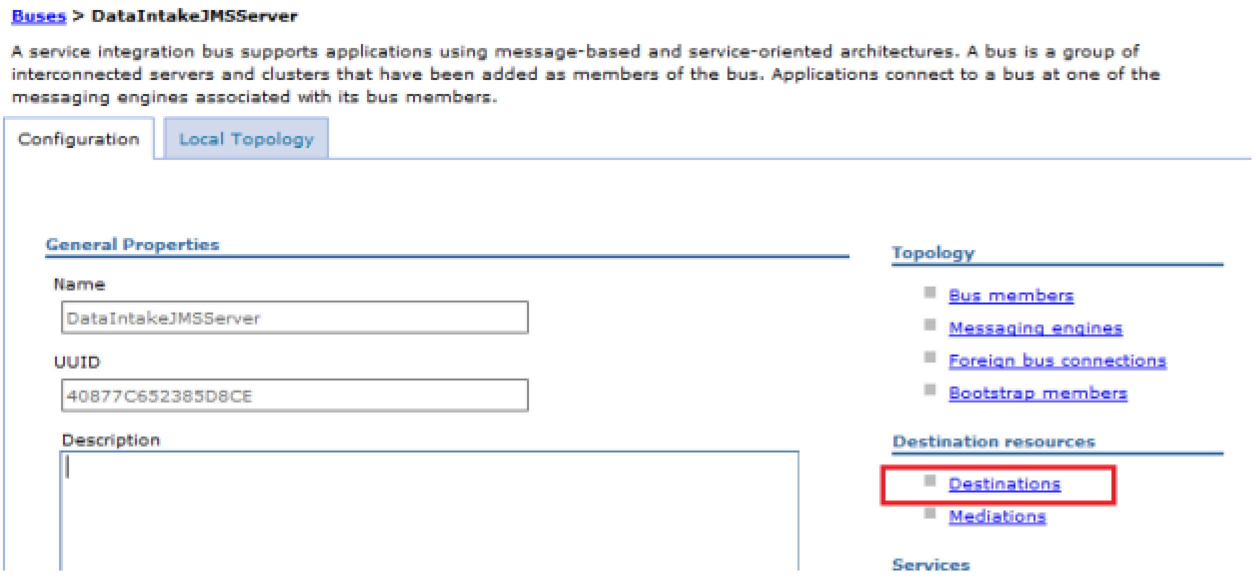
Select	Name ◇	Type ◇	Messaging engine policy assistance ◇
You can administer the following resources:			
□	slc03sueNode01:oipa_cycle	Server	Not applicable
Total 1			

6. Click on **Next > Next > Finish.** After successful creation you should be seeing the bus member i.e added

Configure the destinations

To configure the destinations for messaging, do the following:

1. Select **Service integration > Buses**.
2. Click the DataIntakeJMSServer.
3. Under Destination Resources, click **Destinations**.



Buses > DataIntakeJMSServer

A service integration bus supports applications using message-based and service-oriented architectures. A bus is a group of interconnected servers and clusters that have been added as members of the bus. Applications connect to a bus at one of the messaging engines associated with its bus members.

Configuration | Local Topology

General Properties

Name: DataIntakeJMSServer

UUID: 40877C652385D8CE

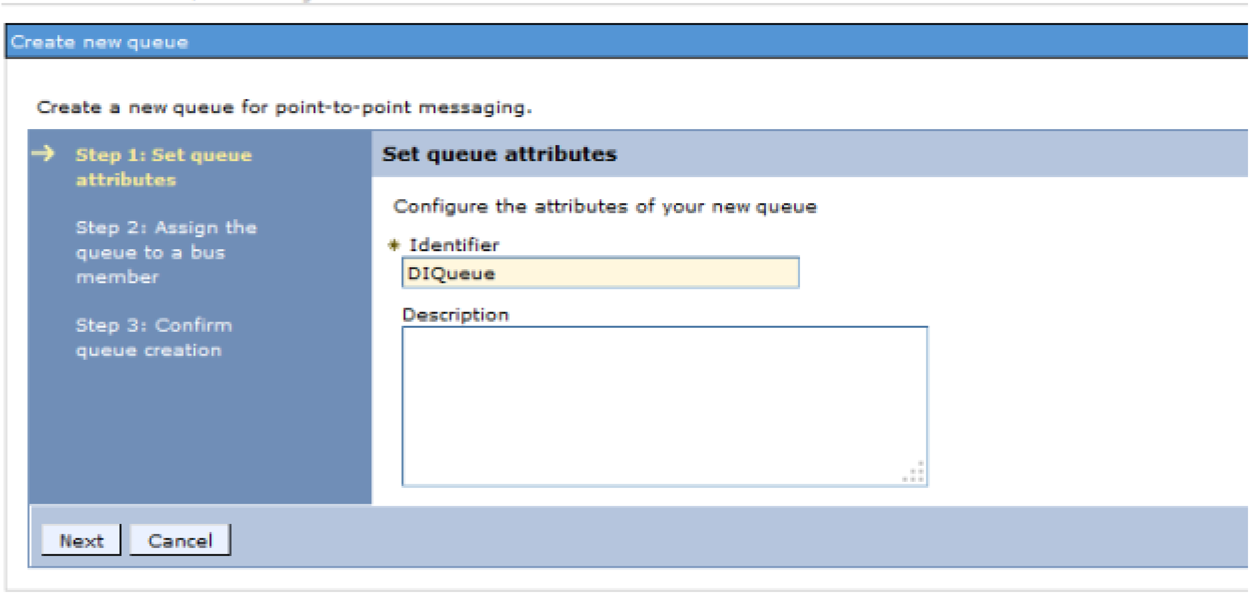
Description:

Topology
- Bus members
- Messaging engines
- Foreign bus connections
- Bootstrap members

Destination resources
- Destinations
- Mediations

Services

4. Click on New and select Queue and give the Queue Name as DIQueue as shown below



Create new queue

Create a new queue for point-to-point messaging.

→ Step 1: Set queue attributes

Step 2: Assign the queue to a bus member

Step 3: Confirm queue creation

Set queue attributes

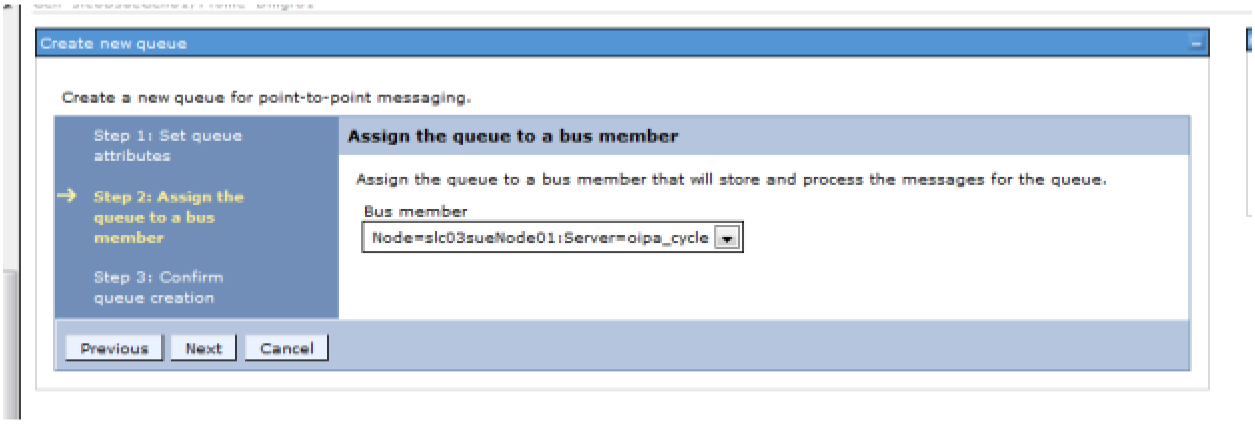
Configure the attributes of your new queue

* Identifier: DIQueue

Description:

Next | Cancel

5. Click on Next and select the Bus member that is created earlier.



Create new queue

Create a new queue for point-to-point messaging.

Step 1: Set queue attributes

→ Step 2: Assign the queue to a bus member

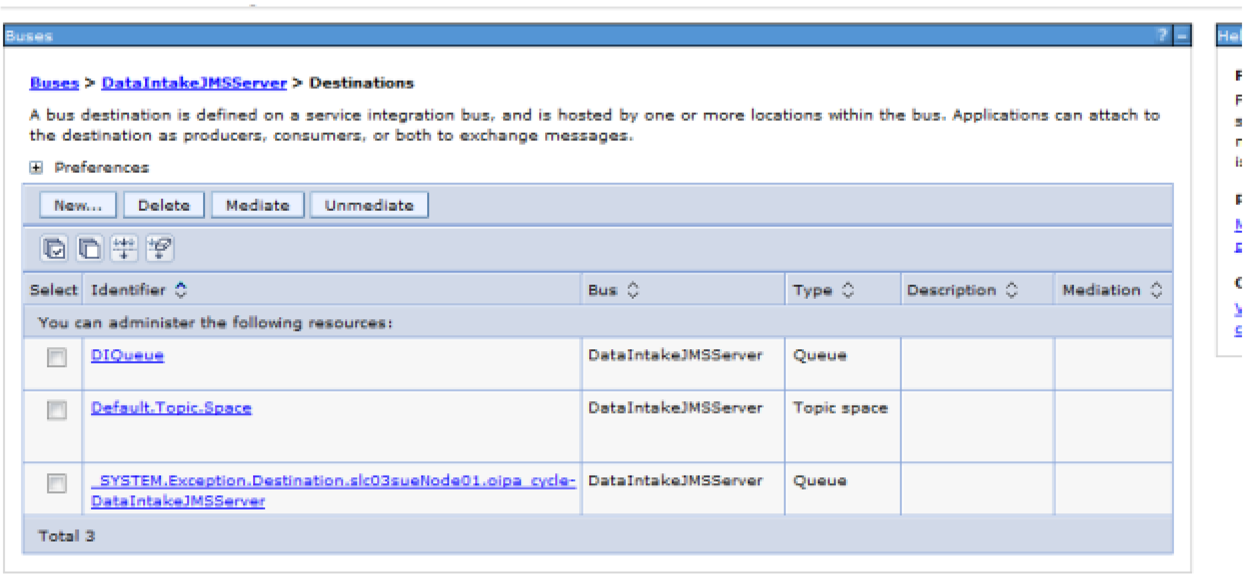
Step 3: Confirm queue creation

Assign the queue to a bus member

Assign the queue to a bus member that will store and process the messages for the queue.

Bus member: Node=slc03sueNode01:Server=oipa_cycle

6. Click on Next > Finish. After succesful creation of Queue you should be seeing it as shown below.



Buses

Buses > DataIntakeJMSServer > **Destinations**

A bus destination is defined on a service integration bus, and is hosted by one or more locations within the bus. Applications can attach to the destination as producers, consumers, or both to exchange messages.

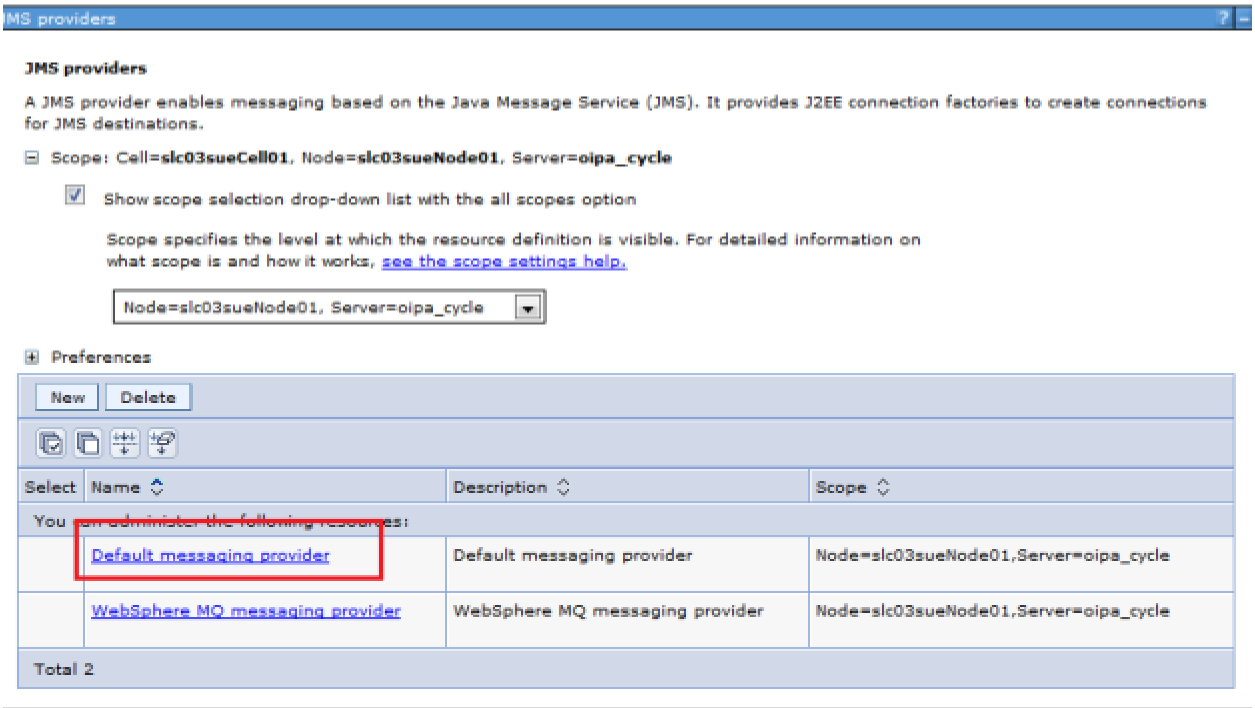
⊞ Preferences

Select	Identifier	Bus	Type	Description	Mediation
You can administer the following resources:					
□	DIQueue	DataIntakeJMSServer	Queue		
□	Default.Topic.Space	DataIntakeJMSServer	Topic space		
□	_SYSTEM.Exception.Destination.slc03sueNode01.oipa_cycle-DataIntakeJMSServer	DataIntakeJMSServer	Queue		
Total 3					

Configure JMS connection queue factory

To configure the JMS connection queue factory, do the following:

1. Display the default messaging provider:
   - **Select Resources > JMS Providers**.
2. In the opened window, select Default Messaging Provider as shown below



MS providers

JMS providers

A JMS provider enables messaging based on the Java Message Service (JMS). It provides J2EE connection factories to create connections for JMS destinations.

⊟ Scope: Cell=**slc03sueCell01**, Node=**slc03sueNode01**, Server=**oipa_cycle**

☑ Show scope selection drop-down list with the all scopes option

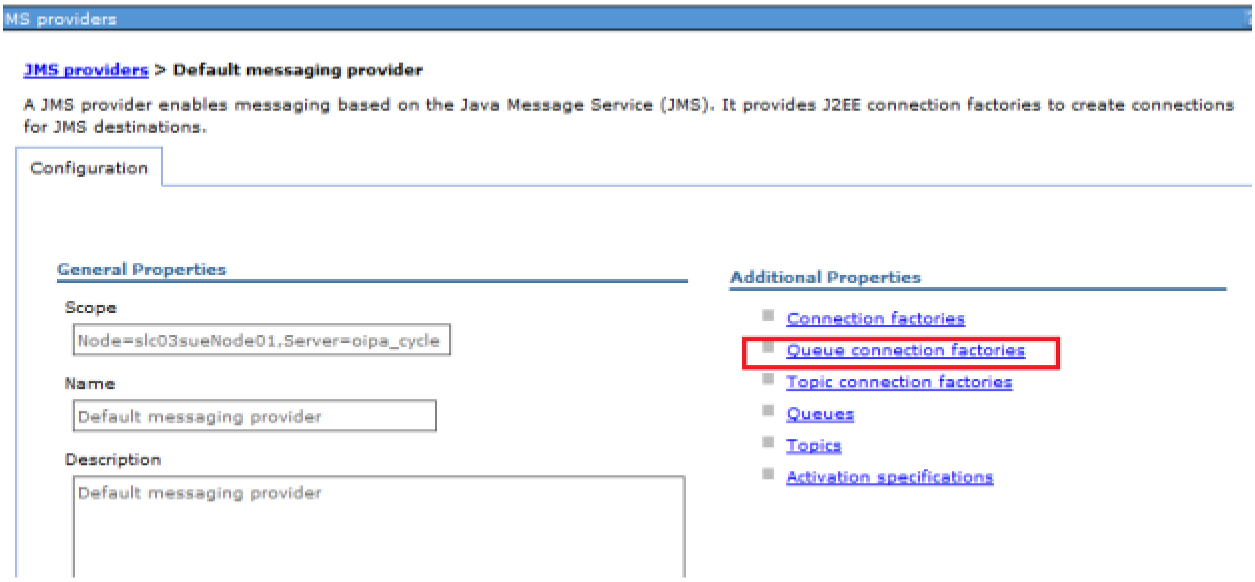
Scope specifies the level at which the resource definition is visible. For detailed information on what scope is and how it works, see the scope settings help.

Node=slc03sueNode01, Server=oipa_cycle

⊞ Preferences

Select	Name	Description	Scope
You can administer the following resources:			
	Default messaging provider	Default messaging provider	Node=slc03sueNode01,Server=oipa_cycle
	WebSphere MQ messaging provider	WebSphere MQ messaging provider	Node=slc03sueNode01,Server=oipa_cycle
Total 2			

3. In the opened window select Queue Connection Factory



MS providers

JMS providers > **Default messaging provider**

A JMS provider enables messaging based on the Java Message Service (JMS). It provides J2EE connection factories to create connections for JMS destinations.

General Properties

Scope
Node=slc03sueNode01,Server=oipa_cycle

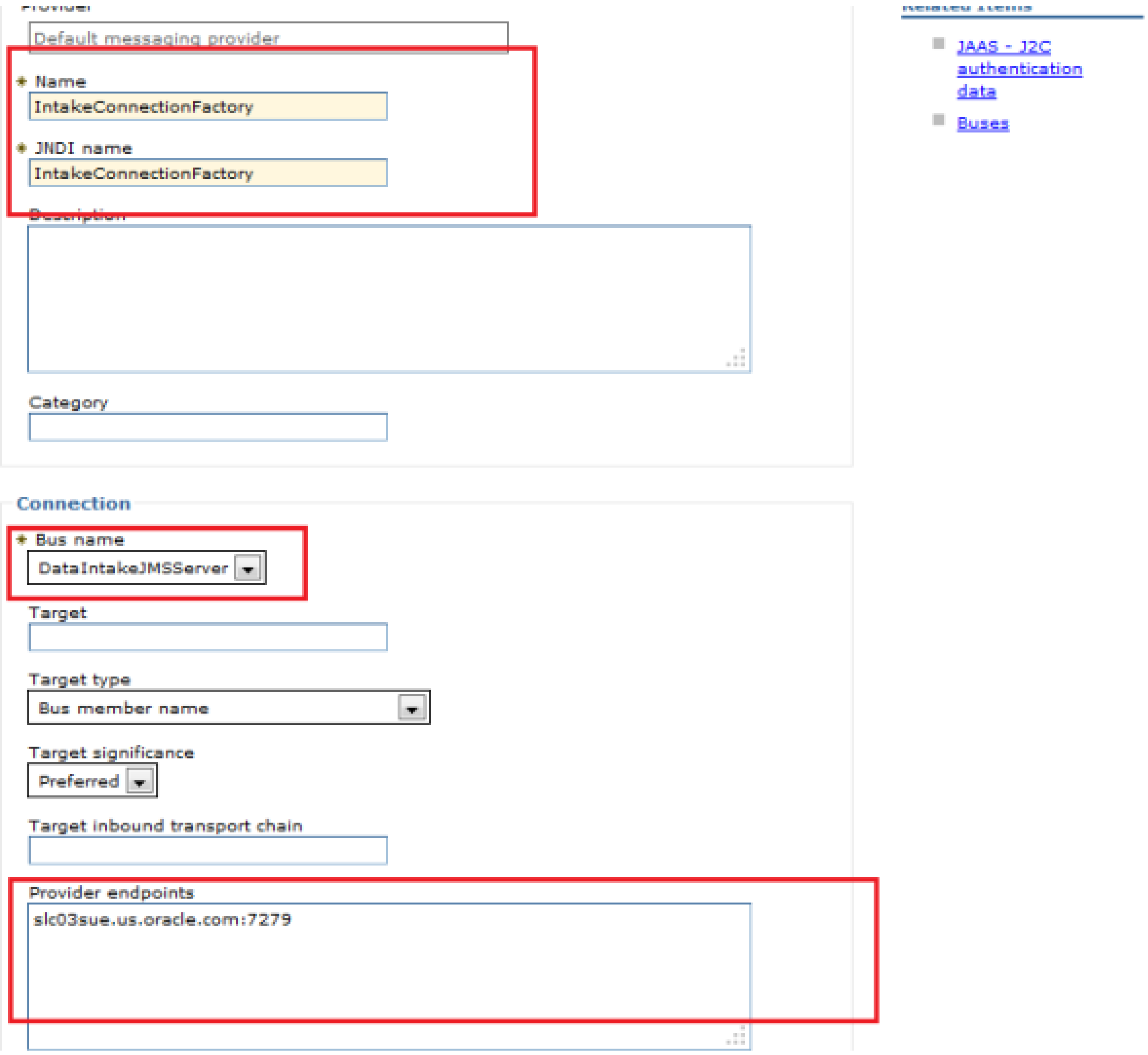
Name
Default messaging provider

Description
Default messaging provider

Additional Properties

- Connection factories
- Queue connection factories
- Topic connection factories
- Queues
- Topics
- Activation specifications

4. Click on **New**
5. Enter the Connection factory Details as
   - **Name** : IntakeConnectionFactory
   - **JNDI Name** : IntakeConnection Factory
   - **Bus** : DataIntakeJMSServer
   - **Provider End Point** : <host> PortNumber
     - port number above will be the server end point address and can be checked at **Server > Ports**



Provider
Default messaging provider

* Name
IntakeConnectionFactory

* JNDI name
IntakeConnectionFactory

Description

Category

Connection

* Bus name
DataIntakeJMSServer

Target

Target type
Bus member name

Target significance
Preferred

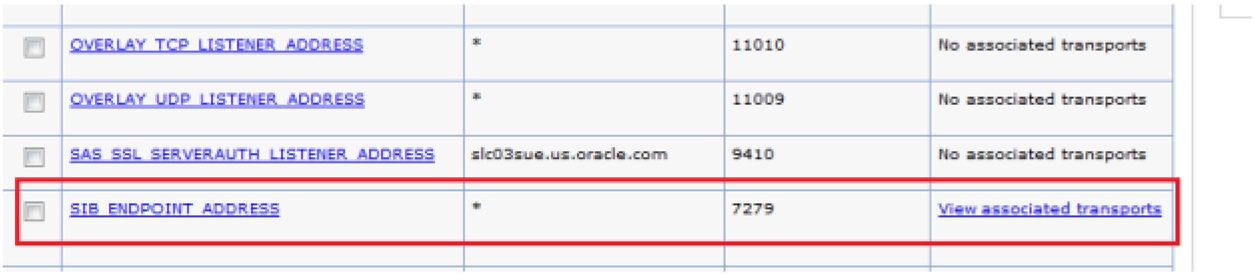
Target inbound transport chain

Provider endpoints
slc03sue.us.oracle.com:7279

Provider Endpoint above will be <host>:<SIB end point port Address>

SIM end point can be seen by clicking on Server > Ports >

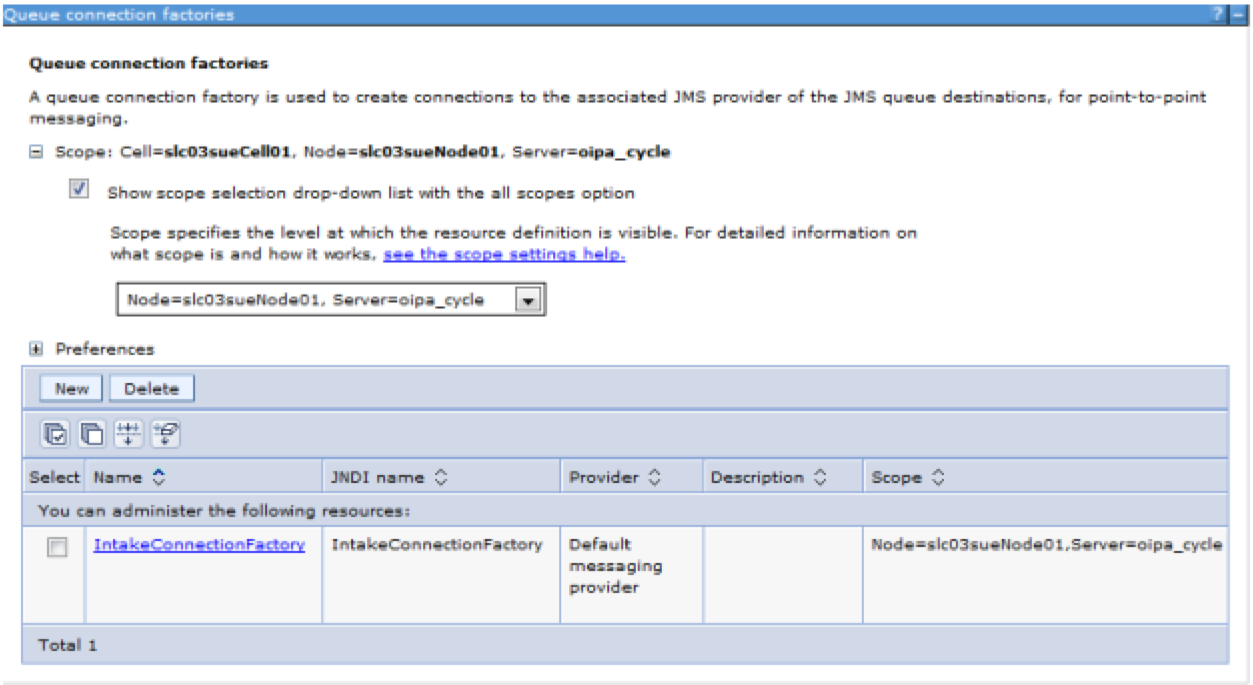
Default will be **localhost:7276**



□	OVERLAY_TCP_LISTENER_ADDRESS	*	11010	No associated transports
□	OVERLAY_UDP_LISTENER_ADDRESS	*	11009	No associated transports
□	SAS_SSL_SERVERAUTH_LISTENER_ADDRESS	slc03sue.us.oracle.com	9410	No associated transports
□	SIB_ENDPOINT_ADDRESS	*	7279	View associated transports

6. Click on **Apply** and save to **master configuration**.

Note: After successful creation of Queue Connection factory, you should be seeing it under Queue Connection factories.



Queue connection factories

Queue connection factories

A queue connection factory is used to create connections to the associated JMS provider of the JMS queue destinations, for point-to-point messaging.

⊟ Scope: Cell=**slc03sueCell01**, Node=**slc03sueNode01**, Server=**oipa_cycle**

☑ Show scope selection drop-down list with the all scopes option

Scope specifies the level at which the resource definition is visible. For detailed information on what scope is and how it works, see the scope settings help.

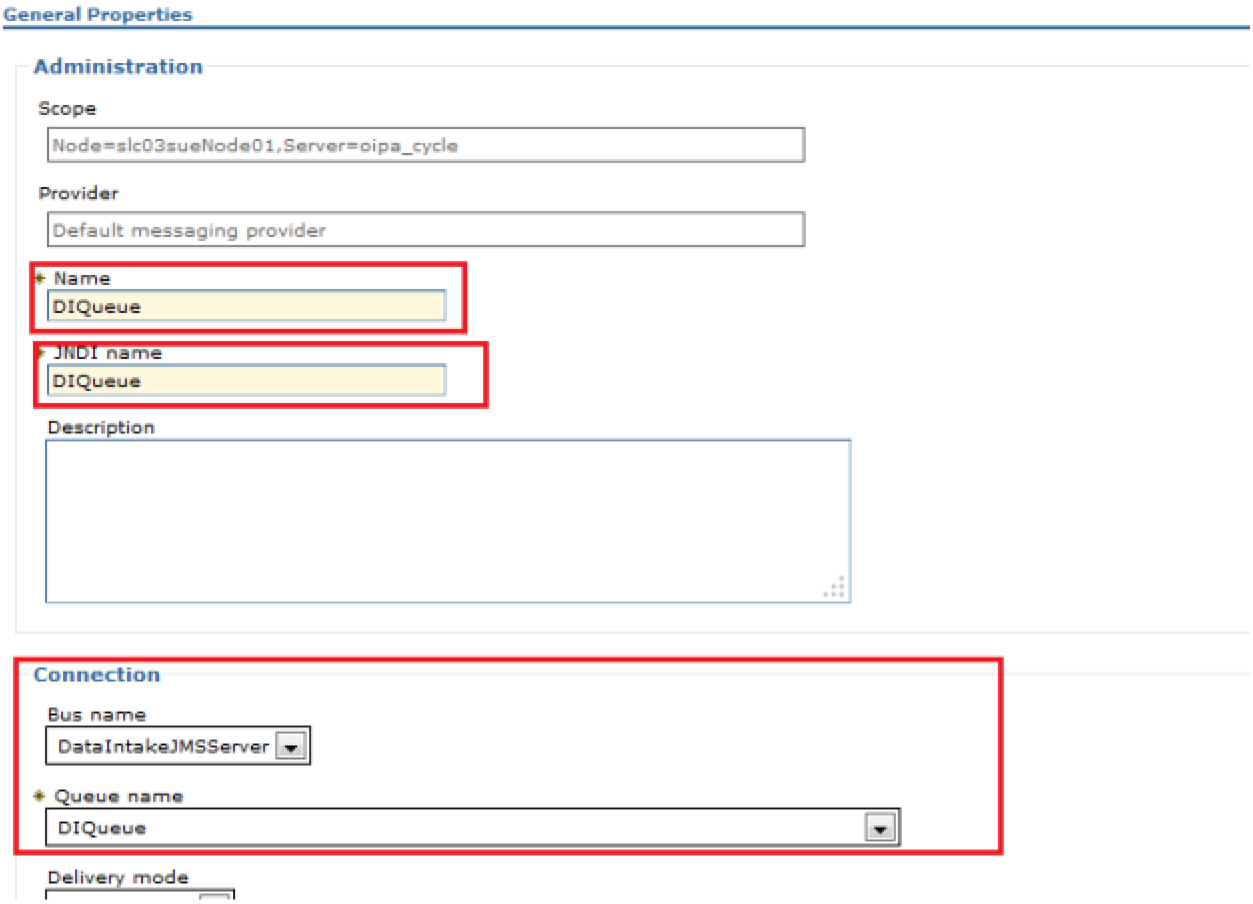
Node=slc03sueNode01, Server=oipa_cycle

⊞ Preferences

Select	Name	JNDI name	Provider	Description	Scope
You can administer the following resources:					
□	IntakeConnectionFactory	IntakeConnectionFactory	Default messaging provider		Node=slc03sueNode01,Server=oipa_cycle
Total 1					

Configure JMS Queue

1. Select **Resources > JMS Providers**
2. Click on **Default messaging provide**r and click on Queues under the Additional Properties. Click on **New**
3. Give the Queue Details as mentioned below
   - **Name:** DIQueue
   - **JNDI Name:** DIQueue
   - **Bus Name:** DataIntakeJMSServer
   - **Queue :** DIQueue
   - Apply and save to master configuration.



General Properties

Administration

Scope: Node=slc03sueNode01,Server=oipa_cycle

Provider: Default messaging provider

* Name: DIQueue

* JNDI name: DIQueue

Description:

Connection

Bus name: DataIntakeJMSServer

* Queue name: DIQueue

Delivery mode

Verify the messaging engine startup

To verify the messaging engine startup, do the following:

1. Ensure the application server has been restarted.
2. Start the **WebSphere Administrative Console.**
3. Select **Service integration > Buses.**
4. Click **DataIntakeJMSServer**.
5. Under **Toplology** click on **Messaging engines.**
6. You should see the green arrow under Status to note the Node.server-DataIntakeJMSServer Messaging Engine has been started.

Update Global Security Settings

In Application Server Admin Console,

1. Go to **Security -> Global Security** (in the left side menu)
2. Under **Authentication panel -> Expand RMI/IIOP security**
3. Click on **CSIv2 inbound communications ->**
   - CSIv2 Transport Layer ->
   - set Transport : SSL-supported
   - Similarly, for CSIv2 outbound communications ->

- CSIv2 Transport Layer ->
- set Transport : SSL-supported

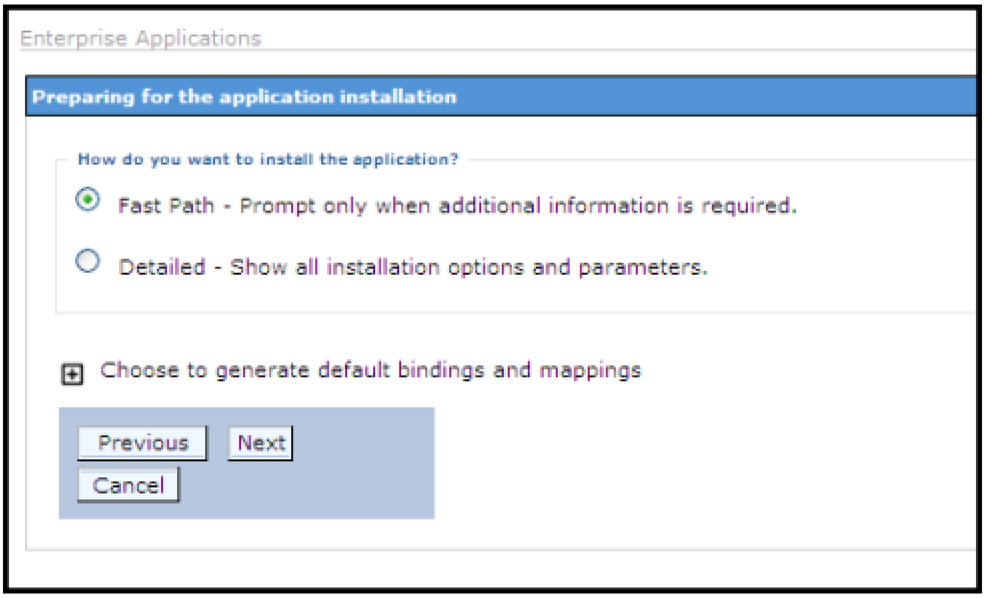
4. Apply and Save the configuration settings. **Restart** the Cycle Server.

Configuration on the DI Client side.

1. Update the below parameters in CycleClient.properties file on the DI client/conf folder
   - contextFactory=com.ibm.websphere.naming.WsnInitialContextFactory
   - provider.url=iiop://slc03sue.us.oracle.com:2812 (port number should match with the Boot strap address on websphere application server ports)
2. The below mentioned jars should be provided on the client side lib directory. These are available in the websphere libraries
   - com.ibm.ws.ejb.thinclient_8.5.0.jar
   - com.ibm.ws.orb_8.5.0.jar
   - com.ibm.ws.sib.client.thin.jms_8.5.0.jar

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.



Enterprise Applications

Preparing for the application installation

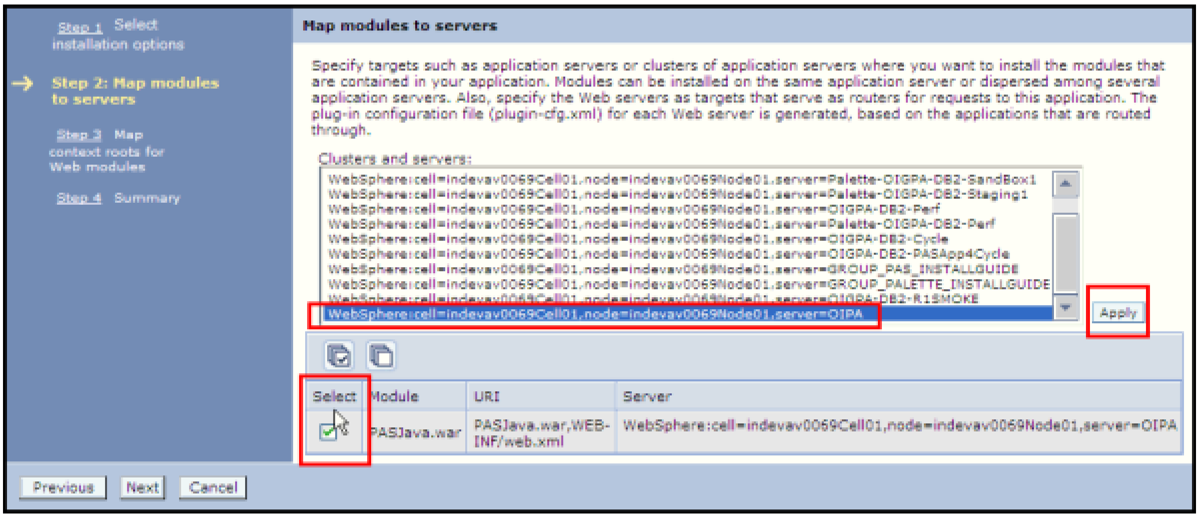
How do you want to install the application?

● Fast Path - Prompt only when additional information is required.

○ Detailed - Show all installation options and parameters.

⊞ Choose to generate default bindings and mappings

6. On the Select installation options screen, click Next.
7. On the Map modules to servers screen:
   a. Select the server from the scrolling list.
   b. Check the box for PASJava.war and click Apply.
   c. Click Next.



Step 1 Select installation options
→ Step 2: Map modules to servers
Step 3 Map context roots for Web modules
Step 4 Summary

Map modules to servers

Specify targets such as application servers or clusters of application servers where you want to install the modules that are contained in your application. Modules can be installed on the same application server or dispersed among several application servers. Also, specify the Web servers as targets that serve as routers for requests to this application. The plug-in configuration file (plugin-cfg.xml) for each Web server is generated, based on the applications that are routed through.

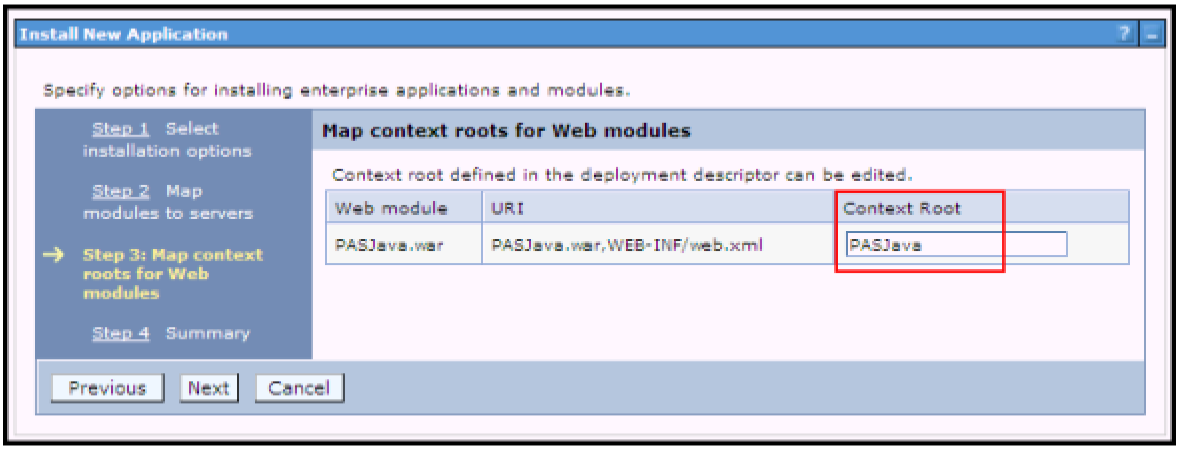
Clusters and servers:

```
WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=Palette-OIGPA-DB2-SandBox1
WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=Palette-OIGPA-DB2-Staging1
WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=OIGPA-DB2-Perf
WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=Palette-OIGPA-DB2-Perf
WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=OIGPA-DB2-Cycle
WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=OIGPA-DB2-PASApp4Cycle
WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=GROUP_PAS_INSTALLGUIDE
WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=GROUP_PALETTE_INSTALLGUIDE
WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=OIGPA-DB2-R1SMOKE
WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=OIPA
```

Select	Module	URI	Server
☑	PASJava.war	PASJava.war,WEB-INF/web.xml	WebSphere:cell=indevav0069Cell01,node=indevav0069Node01,server=OIPA

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**.


Install New Application
Step 1 Select installation options
Step 2 Map modules to servers
Step 3: Map context roots for Web modules
Step 4 Summary
Previous
Next
Cancel

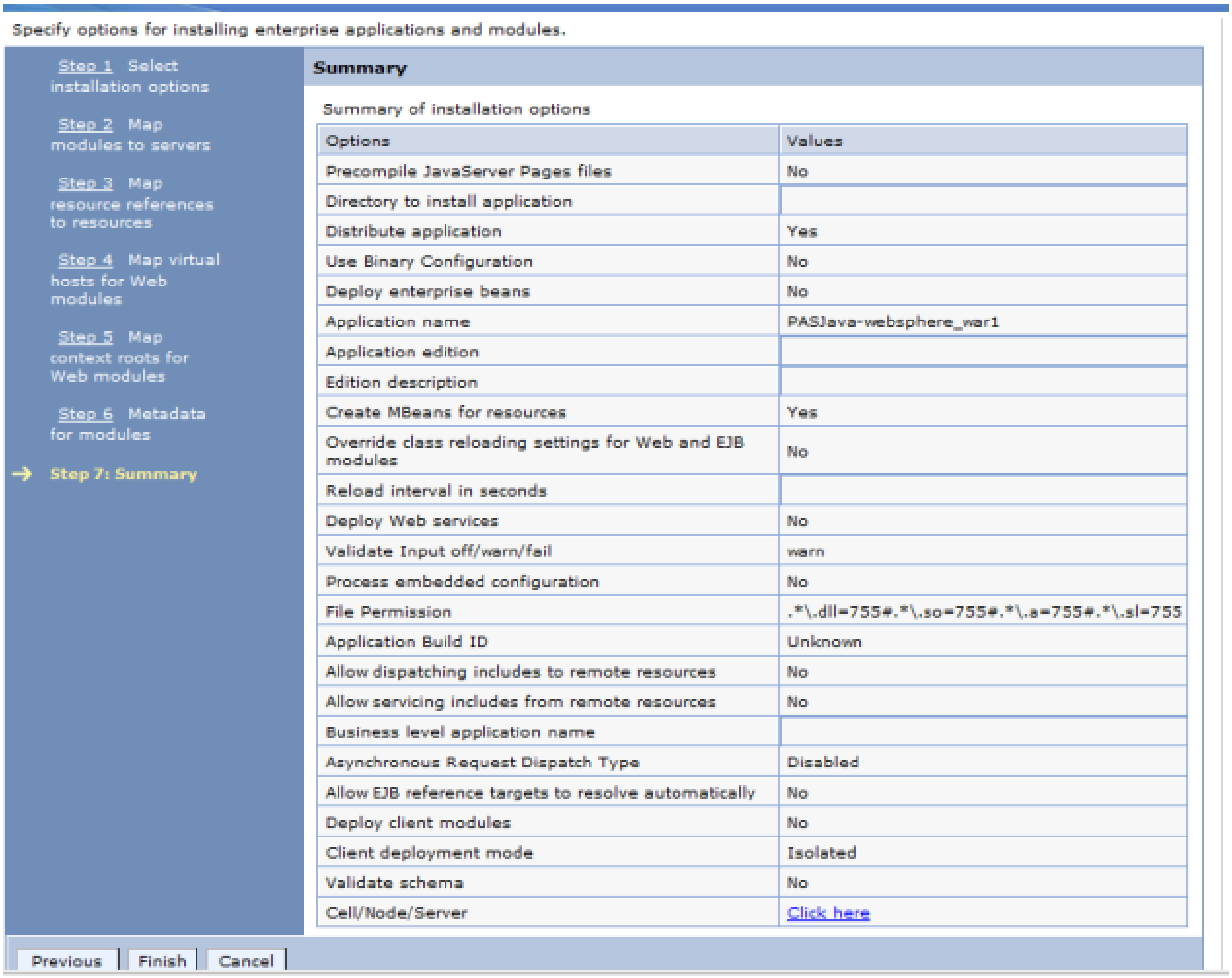
Specify options for installing enterprise applications and modules.

Map context roots for Web modules

Context root defined in the deployment descriptor can be edited.

Web module	URI	Context Root
PASJava.war	PASJava.war,WEB-INF/web.xml	PASJava

11. On the **Metadata for modules** screen, click **Next**.
12. Confirm the settings on the Summary screen and click **Finish**.


Step 1 Select 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
Previous
Finish
Cancel

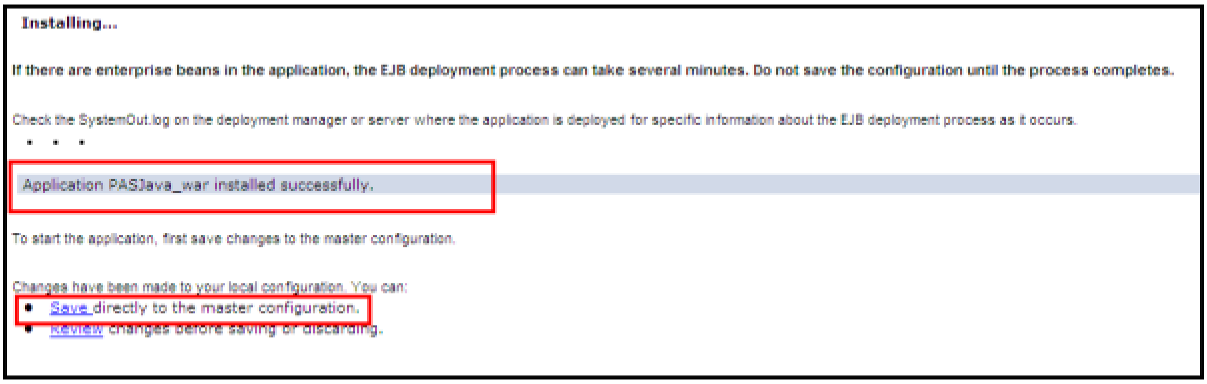
Specify options for installing enterprise applications and modules.

Summary

Summary of installation options

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

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:

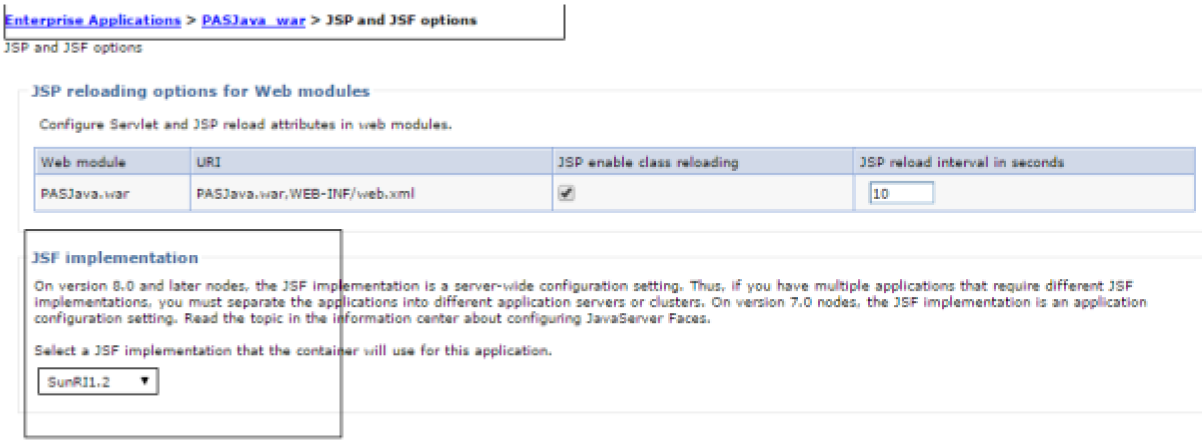
- Save directly to the master configuration.
- Review changes before saving or discarding.

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-3.5.2.jar
   - aspectjrt-1.8.7.jar
   - aspectjweaver-1.8.7.jar
   - commons-collections-3.2.2.jar
   - commons-logging-1.2.jar
   - log4j-1.2.17.jar
   - coherence.jar
   - oraclepki.jar
   - osdt_cert.jar
   - osdt_core.jar
   - javax.ws.rs-api-2.0.1.jar
6. For the Native Library Path field, enter the full path for the following jar file:
   - el-api-2.2.jar
7. Check "Use an isolated class loader for this shared library" under class loading and 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
17. Select **Applications>Application Types>WebSphere** enterprise application.
18. Select the **PASJava.war** application.
19. Click **JSP** and **JSF** options under Web Module Properties and select the JSF implementation to **SunRI1.2**



Enterprise Applications > PASJava_war > JSP and JSF options

JSP and JSF options

JSP reloading options for Web modules

Configure Servlet and JSP reload attributes in web modules.

Web module	URI	JSP enable class reloading	JSP reload interval in seconds
PASJava.war	PASJava.war,WEB-INF/web.xml	☑	10

JSF implementation

On version 8.0 and later nodes, the JSF implementation is a server-wide configuration setting. Thus, if you have multiple applications that require different JSF implementations, you must separate the applications into different application servers or clusters. On version 7.0 nodes, the JSF implementation is an application configuration setting. Read the topic in the information center about configuring JavaServer Faces.

Select a JSF implementation that the container will use for this application.

SunRI1.2 ▼

20. Under Detailed Properties, click **Class loading and update detection**.
21. Select **Classes loaded with local class loader first (parent last)** option.
22. Select **Class loader for each WAR file in application** option.
23. Click **OK**.
24. 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."
5. Create and Configure the Web Application Utility.

Using a web browser, connect to the Administrative Console using the appropriate server_name and port. (Ex: server_name:port/admin" class="external-link" rel="nofollow">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.

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

   > **Note:** Be sure to include the trailing slash (/ or) character.
6. Set **Initial Heap Size** to **256**.
7. Set **Maximum Heap Size** to **512**.
8. 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**.

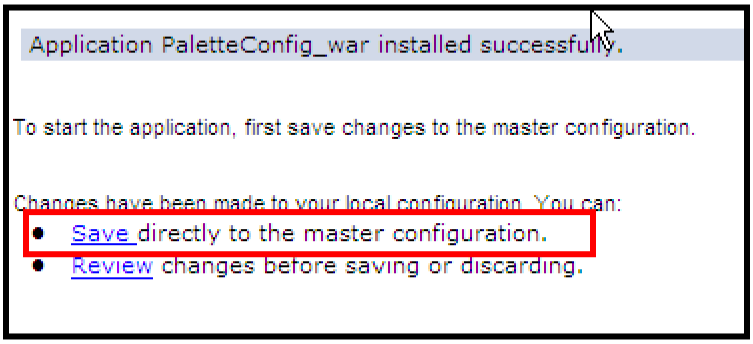
Configure the Virtual Host

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:
   a. Select the server from the scrolling list.
   b. Check the box for PaletteConfig.war and click **Apply**
   c. 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.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, uncheck the 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.



Application PaletteConfig_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.

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). Be sure to supply the appropriate drivers for the database type you are using.
   - AIX or Linux example for Oracle:

   `/opt/oracle/db_drivers/ojdbc7-12.1.0.1.jar`

   - Windows example for Oracle:

   `C:\oracle\db_drivers\ojdbc7-12.1.0.1.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.