

Oracle® Healthcare Master Person Index

WebLogic User's Guide

Release 2.0.2

E25250-02

August 2012

Copyright © 2011, 2012, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

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.

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 RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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 fail-safe, 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.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

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.

Contents

Preface	v
Audience	v
Documentation Accessibility	v
Related Documents	v
Finding Information and Patches on My Oracle Support	vi
Finding Oracle Documentation	viii
Conventions	viii
1 Getting Started With Oracle WebLogic Server	
Creating an Instance of an Oracle WebLogic Server	1-1
Related Documentation	1-1
Oracle WebLogic Server Requirements	1-1
Creating a Project and Starting the OHMPI Wizard	1-2
To Create a Project and Start the OHMPI Wizard	1-2
Adding an Oracle WebLogic Server	1-2
To Add an Oracle WebLogic Server	1-2
Creating, Configuring, and Building a Master Person Index Application	1-5
2 Oracle WebLogic Server Configuration	
Configuring an Oracle WebLogic Server	2-1
Configuring the HL7 v2 Server (only for IHE Profiles Application)	2-1
To Install the WebLogic Application Lifecycle Listener	2-2
Configuring the Audit Client (only for IHE Profiles Application)	2-2
Integrating TopLink Essentials with Oracle WebLogic Server (only for IHE Profiles Application)	2-2
Installing a Database Driver for MySQL	2-3
To Install a Database Driver	2-3
Creating JDBC Data Resources for an IHE Profile Application Project	2-3
To Create JDBC Data Resources for an IHE Profiles Application Project for MySQL	2-3
To Create JDBC Data Resources for an IHE Profiles Application Project for Oracle	2-7
Creating JMS Resources for an IHE Profile Application Project	2-8
To Create JMS Server	2-8
To Create JMS Module	2-10
To Create JMS Connection Factory	2-11
To Create JMS Topic	2-12

To Create PixUpdateNotificationTopic	2-13
Creating JDBC Data Resources for an MPI Application Project	2-13
To Create JDBC Data Resources for an MPI Application Project for MySQL	2-13
To Create JDBC Data Resources for an MPI Application Project for Oracle.....	2-17
Creating JMS Resources for an MPI Application Project	2-18
To Create JMS Server.....	2-18
To Create JMS Module	2-20
To Create JMS Connection Factory	2-21
To Create JMS Topic	2-21
Setting Up the User	2-22

3 Using MPI and IHE Profile Applications on WebLogic

Deploying and Running Applications on Oracle WebLogic Server	3-1
To Deploy and Run Applications on an Oracle WebLogic Server	3-1
Web Layer Separation for OHMPI.....	3-3
Building the Project.....	3-3
Configuring the Application Server	3-3
Configuring Connectivity	3-4
Deploying the Build.....	3-4
WebLogic Clustering	3-4

Preface

Oracle WebLogic Server is a Java Platform, Enterprise Edition (Java EE), application server that supports the deployment the Oracle Healthcare Master Person Index (OHMPI) applications in a runtime environment on a multitude of operating systems. This user's guide provides the information necessary to create an instance of an Oracle WebLogic Server, configure the WebLogic Server, and then deploy and run an MPI Application or an IHE Profile Application on the WebLogic Server.

Audience

This document is intended for Oracle Healthcare Master Person Index users that intend to use an MPI Application or an IHE Profile Application with the Oracle Weblogic Server.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

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.

Related Documents

For more information and instructions for implementing and using a master person index application, see the following documents in the Oracle Healthcare Master Person Index Release 2.0.2 documentation set:

- *Oracle Healthcare Master Person Index Installation Guide*
- *Oracle Healthcare Master Person Index User's Guide*
- *Oracle Healthcare Master Person Index Data Manager User's Guide*
- *Oracle Healthcare Master Person Index Configuration Guide*
- *Oracle Healthcare Master Person Index Configuration Reference*
- *Oracle Healthcare Master Person Index Working With IHE Profiles User's Guide*

- *Oracle Healthcare Master Person Index WebLogic User's Guide*
- *Oracle Healthcare Master Person Index Standardization Engine Reference*
- *Oracle Healthcare Master Person Index Match Engine Reference*
- *Oracle Healthcare Master Person Index Provider Index User's Guide*
- *Oracle Healthcare Master Person Index United States Patient Solution User's Guide*
- *Oracle Healthcare Master Person Index Australia Patient Solution User's Guide*
- *Oracle Healthcare Master Person Index United Kingdom Patient Solution User's Guide*
- *Oracle Healthcare Master Person Index Message Processing Reference*
- *Oracle Healthcare Master Person Index Analyzing and Cleansing Data User's Guide*
- *Oracle Healthcare Master Person Index Loading the Initial Data Set User's Guide*
- *Oracle Healthcare Master Person Index Command Line Reports and Database Maintenance User's Guide*
- *Oracle Healthcare Master Person Index Release Notes*

Finding Information and Patches on My Oracle Support

Your source for the latest information about Oracle Healthcare Master Person Index is Oracle Support's self-service Web site My Oracle Support (formerly MetaLink).

Before you install and use Oracle Healthcare Master Person Index, always visit the My Oracle Support Web site for the latest information, including alerts, White Papers, installation verification (smoke) tests, bulletins, and patches.

Creating a My Oracle Support Account

You must register at My Oracle Support to obtain a user name and password account before you can enter the Web site.

To register for My Oracle Support:

1. Open a Web browser to <https://support.oracle.com>.
2. Click the **Register here** link to create a My Oracle Support account. The registration page opens.
3. Follow the instructions on the registration page.

Signing In to My Oracle Support

To sign in to My Oracle Support:

1. Open a Web browser to <https://support.oracle.com>.
2. Click **Sign In**.
3. Enter your user name and password.
4. Click **Go** to open the My Oracle Support home page.

Finding Information on My Oracle Support

There are many ways to find information on My Oracle Support.

Searching by Article ID

The fastest way to search for information, including alerts, White Papers, installation verification (smoke) tests, and bulletins is by the article ID number, if you know it.

To search by article ID:

1. Sign in to My Oracle Support at <https://support.oracle.com>.
2. Locate the Search box in the upper right corner of the My Oracle Support page.
3. Click the sources icon to the left of the search box, and then select **Article ID** from the list.
4. Enter the article ID number in the text box.
5. Click the magnifying glass icon to the right of the search box (or press the Enter key) to execute your search.

The Knowledge page displays the results of your search. If the article is found, click the link to view the abstract, text, attachments, and related products.

Searching by Product and Topic

You can use the following My Oracle Support tools to browse and search the knowledge base:

- **Product Focus** — On the Knowledge page under Select Product, type part of the product name and the system immediately filters the product list by the letters you have typed. (You do not need to type "Oracle.") Select the product you want from the filtered list and then use other search or browse tools to find the information you need.
- **Advanced Search** — You can specify one or more search criteria, such as source, exact phrase, and related product, to find information. This option is available from the **Advanced** link on almost all pages.

Finding Patches on My Oracle Support

Be sure to check My Oracle Support for the latest patches, if any, for your product. You can search for patches by patch ID or number, or by product or family.

To locate and download a patch:

1. Sign in to My Oracle Support at <https://support.oracle.com>.
2. Click the **Patches & Updates** tab. The Patches & Updates page opens and displays the Patch Search region. You have the following options:
 - In the **Patch ID or Number** field, enter the number of the patch you want. (This number is the same as the primary bug number fixed by the patch.) This option is useful if you already know the patch number.
 - To find a patch by product name, release, and platform, click the **Product or Family** link to enter one or more search criteria.
3. Click **Search** to execute your query. The Patch Search Results page opens.
4. Click the patch ID number. The system displays details about the patch. In addition, you can view the Read Me file before downloading the patch.
5. Click **Download**. Follow the instructions on the screen to download, save, and install the patch files.

Finding Oracle Documentation

The Oracle Web site contains links to all Oracle user and reference documentation. You can view or download a single document or an entire product library.

Finding Oracle Health Sciences Documentation

To get user documentation for Oracle Health Sciences applications, go to the Oracle Health Sciences documentation page at:

<http://www.oracle.com/technetwork/documentation/hsgbu-154445.html>

Note: Always check the Oracle Health Sciences Documentation page to ensure you have the latest updates to the documentation.

Finding Other Oracle Documentation

To get user documentation for other Oracle products:

1. Go to the following Web page:

<http://www.oracle.com/technology/documentation/index.html>

Alternatively, you can go to <http://www.oracle.com>, point to the Support tab, and then click **Documentation**.

2. Scroll to the product you need and click the link.
3. Click the link for the documentation you need.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Getting Started With Oracle WebLogic Server

This chapter provides the procedures for creating an instance of an Oracle WebLogic Server. It includes creating an Oracle Healthcare Master Person Index (OHMPI) project; starting the OHMPI wizard; adding the WebLogic Server; and a listing of where instructions for creating, configuring, and building a Master Person Index application are located.

This chapter includes the following section:

- [Creating an Instance of an Oracle WebLogic Server](#) on page 1

Creating an Instance of an Oracle WebLogic Server

If you are deploying OHMPI on Oracle Business Intelligence Enterprise Edition server, ensure that the OBIEE deployment domain (bifoundation_domain) and the OHMPI project deployment domain run on different ports.

The OHMPI wizard provides a simple and transparent method for you to create your master person index application on the Oracle WebLogic Server. You have to add an Oracle WebLogic Server Instance or select an Oracle WebLogic Server Instance if you already have one or more instances. This process is divided into three individual steps, listed below.

- [Creating a Project and Starting the OHMPI Wizard](#) on page 2
- [Adding an Oracle WebLogic Server](#) on page 2
- [Creating, Configuring, and Building a Master Person Index Application](#) on page 5

Related Documentation

For additional information on the Oracle WebLogic Server, go to the Oracle WebLogic Server Documentation Library, 11g Release 1 (11.1.1) at http://download.oracle.com/docs/cd/E14571_01/wls.htm.

Oracle WebLogic Server Requirements

The Oracle WebLogic Server required environments include:

- NetBeans IDE 6.9.1
- JDK 1.6.0_20 and later
- WebLogic 11gR1 and later

Creating a Project and Starting the OHMPI Wizard

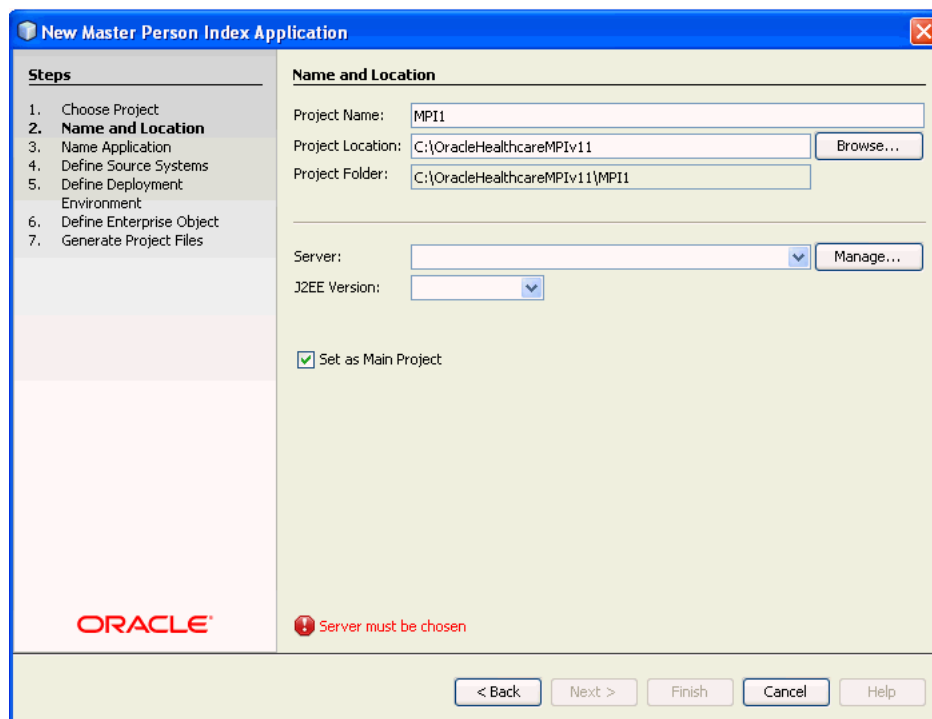
This section documents the steps to create an OHMPI project and start the OHMPI wizard.

To Create a Project and Start the OHMPI Wizard

1. On the NetBeans toolbar, click **New Project**.
2. Under Categories, choose **OHMPI**.
3. Under Projects, choose **Master Person Index Application** and then click **Next**.
4. Type the project name and the path where you want to store the project files in the upper portion of the window.

The New Master Person Index Application page appears.

Figure 1–1 New Master Person Index Application Page



Adding an Oracle WebLogic Server

This section provides the steps for adding an instance of an Oracle WebLogic Server.

To Add an Oracle WebLogic Server

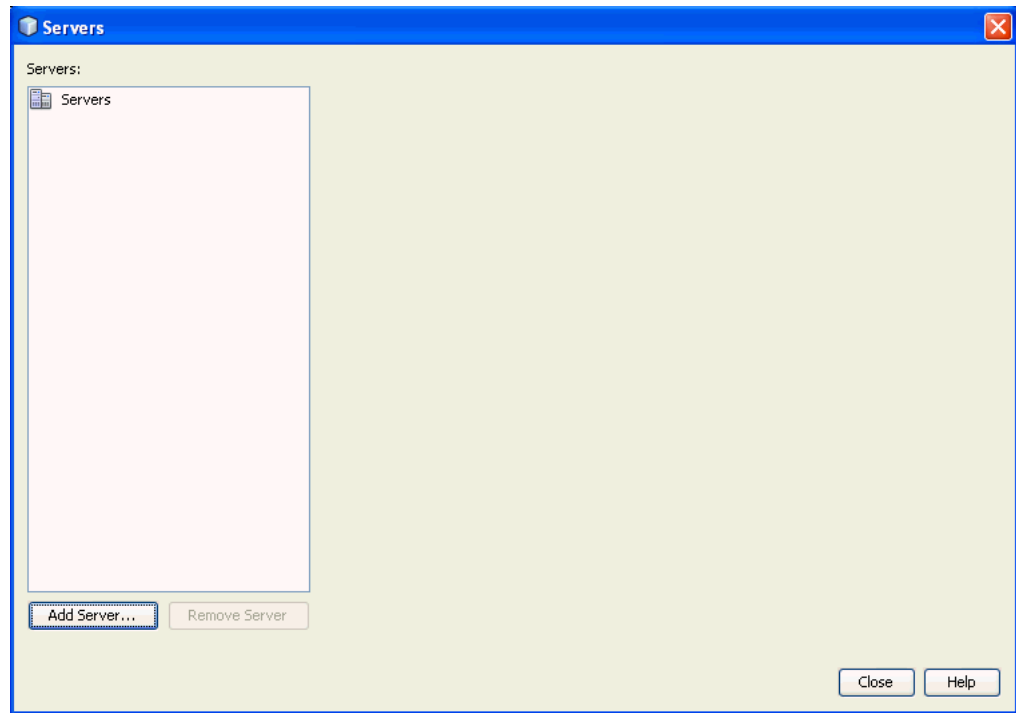
1. If an Oracle WebLogic Server instance is not listed in the Server field drop-down list, click **Manage** to add it.

Note: If an Oracle WebLogic Server instance is listed in the Server field, or if you can select it from the drop-down list, click **Next** to continue to [Creating, Configuring, and Building a Master Person Index Application](#).

The Servers page appears (see Figure 1-2, below) without the WebLogic Server listed under Servers.

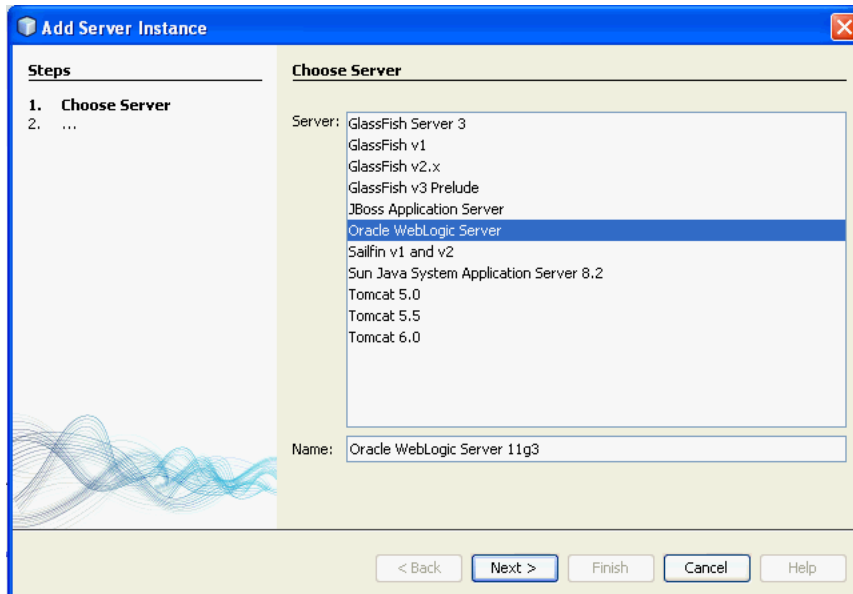
Note: If there is a server listed, its properties display on the right side of the page.

Figure 1-2 Servers Page



2. On the Servers page, click **Add Server**.
The Add Server Instance page appears.

Figure 1–3



3. In the Server list in the upper portion of the Choose Server panel, choose **Oracle WebLogic Server**.
4. In the Name field in the lower portion of the Choose Server panel, type the instance name of your Oracle WebLogic Server (for example: Oracle WebLogic Server 11gR1).
5. Click **Next**.

The Add Server Instance page moves to the second step.

6. Under Server Location in the upper portion of the window, browse to the root location of your Oracle WebLogic Server instance.

Note: Optionally, you can type the root location of your Oracle WebLogic Server instance (for example, C:\Oracle\Weblogic11gR1\wlserver_10.3).

7. Click **Next**.

The Add Server Instance page moves to the third step, Instance Properties. The fields have been populated except for the password.

8. Type a password for your Oracle WebLogic Server Instance.

Note: Do not change the Instance Properties that were populated for you.

9. Click **Finish**.

The Servers page reappears with the WebLogic Server you created listed on the left side of the page. Its properties are displayed on the right side of the page.

Note: If there is more than one server in the list, select the server you created to display its properties.

10. Click Close.

The name of the Oracle WebLogic Server instance you defined should appear in the Server field under Name and Location on the right side of the New Master Index Application page.

11. Do one of the following:

- If the Oracle WebLogic Server instance is listed in the Server field, click **Next**.
- If the Oracle WebLogic Server instance is not listed in the Server field, select it from the drop-down list and click **Next**.

You are now ready to move on to the third step of this process, [Creating, Configuring, and Building a Master Person Index Application](#).

Creating, Configuring, and Building a Master Person Index Application

The generic instructions to create, configure, and build a master person index application are documented elsewhere. To perform these tasks see:

- *Oracle Healthcare Master Person Index User's Guide*
- *Oracle Healthcare Master Person Index Configuration Guide*
- *Oracle Healthcare Master Person Index User's Guide*

There is nothing for you to do as there are no specific procedures that are relevant to the Oracle WebLogic Server that you need to perform. The OHMPI wizard automatically generates and packages the artifacts of Oracle WebLogic Server for you.

Oracle WebLogic Server Configuration

This chapter provides procedures for configuring an Oracle WebLogic Server, including installing a database driver, creating JDBC data resources, creating JMS resources, and setting up a user.

This chapter includes the following section:

- [Configuring an Oracle WebLogic Server](#) on page 1

Configuring an Oracle WebLogic Server

Before using your Oracle WebLogic Server you must prepare it to host your Master Person Index (MPI) Application or your IHE Profiles Application, using the following procedures:

- [Configuring the HL7 v2 Server \(only for IHE Profiles Application\)](#) on page 1
- [Configuring the Audit Client \(only for IHE Profiles Application\)](#) on page 2
- [Integrating TopLink Essentials with Oracle WebLogic Server \(only for IHE Profiles Application\)](#) on page 2
- [Installing a Database Driver for MySQL](#) on page 3
- Specific to an IHE Profiles Application project
 - [Creating JDBC Data Resources for an IHE Profile Application Project](#) on page 3
 - [Creating JMS Resources for an IHE Profile Application Project](#) on page 8
- Specific to an MPI Application project
 - [Creating JDBC Data Resources for an MPI Application Project](#) on page 13
 - [Creating JMS Resources for an MPI Application Project](#) on page 18
- [Setting Up the User](#) on page 22

Configuring the HL7 v2 Server (only for IHE Profiles Application)

1. After building an IHE Profiles Application project, unzip **hl7v2.zip** to a directory of your choice (for example, your OHMPI installation directory).
2. Set the **OHMPI_IHE_HL7V2_HOME** environment variable to the unzipped **hl7v2** directory from the previous step.
3. Follow the existing instructions to configure and deploy the IHE Profiles Application to the WebLogic Server.

Note: The rest of the HL7 v2 server configuration and the SSL configuration for WebLogic is the same as for GlassFish. For detailed information, see *Oracle Healthcare Master Person Index Working With IHE Profiles User's Guide*.

To Install the WebLogic Application Lifecycle Listener

You do not have to install the WebLogic Application Lifecycle Listener as it is automatically installed when you deploy an IHE Profiles Application.

Configuring the Audit Client (only for IHE Profiles Application)

When you build the IHE Profiles Application project for WebLogic, five artifacts are created:

- *Project_Name*.ear: The IHE Profiles Application.
- *hl7v2*.zip: The HL7 v2 lifecycle module.

The *hl7v2.zip* is an HL7 v2 server package that includes all artifacts required to run the IHE OHMPI HL7 v2 server and HL7 v2 lifecycle model which manages the server. You unzip it to the location where you run the server. The unzipped folder includes:

hl7v2

– *config*

The *config* sub folder includes all the configuration files.

– *lib*

The *lib* sub folder includes all the required jar files.

- *audit-repo-syslog-client.jar*
- *ihe-atna-audit-client.jar*
- *ohmpi_audit_client.properties*

After building a WebLogic-targeted IHE Profiles Application project (and before starting WebLogic Application Server), manually copy the following three files from the IHE Profiles Application project's *dist* directory to the appropriate WebLogic domain's *lib* directory (for example, *user_projects\domains\domain_name\lib*):

- *audit-repo-syslog-client.jar*: The JAR file that contains the functionality of the audit repository syslog client.
- *ihe-atna-audit-client.jar*: The JAR file that contains the functionality of the IHE ATNA audit client.
- *ohmpi_audit_client.properties*: The property file used to configure where the audit server is located.

Integrating TopLink Essentials with Oracle WebLogic Server (only for IHE Profiles Application)

To integrate TopLink Essentials with Oracle WebLogic Server, do the following:

1. Find the following two TopLink Essentials' jar files under the GlassFish installation's *lib* directory, for example, *OHMPI_HOME\glassfish\lib*:
 - *toplink-essentials.jar*

- toplink-essentials-agent.jar
2. Copy the two jar files into WebLogic domain's lib directory (for example, user_projects\domains\domain_name\lib).

Installing a Database Driver for MySQL

The section provides instructions for installing a database driver if you are using a non-Oracle database.

To Install a Database Driver

1. Find your database driver in your database installation (for example: mysql-connector-java-5.1.12-bin.jar for MySQL).
2. Copy your database driver to your Oracle WebLogic Server wlserver_10.3\server\lib.

Creating JDBC Data Resources for an IHE Profile Application Project

This section provides instructions for creating the JDBC data resources and defining the JDBC connections for an IHE Profile Application Project.

To Create JDBC Data Resources for an IHE Profiles Application Project for MySQL

1. For instructions on how to start and stop your Oracle WebLogic Server, see *Starting and Stopping Servers: Quick Reference* at http://download.oracle.com/docs/cd/E14571_01/wls.htm.
2. Launch the **Oracle WebLogic Server Administration Console**.
3. Log in using the default user Name (**weblogic**) and Password (**welcome1**).
The Oracle WebLogic Administration Console appears.
4. On the left panel, under Domain Structure, expand **Services**, click **Data Sources**.

Figure 2–1 WebLogic Administration Console - Summary of JDBC Data Sources

Summary of JDBC Data Sources

A JDBC data source is an object bound to the JNDI tree that provides database connectivity through a pool of JDBC connections. Applications can look up a data source on the JNDI tree and then borrow a database connection from a data source.

This page summarizes the JDBC data source objects that have been created in this domain.

[▶ Customize this table](#)

Data Sources (Filtered - More Columns Exist)

Showing 1 to 3 of 3 Previous | Next

<input type="checkbox"/>	Name	JNDI Name	Targets
<input type="checkbox"/>	PatientDataSource	jdbc/PatientDataSource	AdminServer
<input type="checkbox"/>	PatientSequenceDataSource	jdbc/PatientSequenceDataSource	AdminServer
<input type="checkbox"/>	PIXDomainLUDDataSource	jdbc/PIXDomainLUDDataSource	AdminServer

Showing 1 to 3 of 3 Previous | Next

A Summary of JDBC Data Sources appears in the right panel.

5. To create a new JDBC Data Source click **New** at the bottom of the right panel and select **Generic Data Source** from the drop down list.

Settings for a new JDBC Data Source appears in the right panel of the page. It is here that you will create a new JDBC Data Source.

Figure 2–2 Create a New JDBC Data Source Panel

Create a New JDBC Data Source

Back | Next | Finish | Cancel

JDBC Data Source Properties

The following properties will be used to identify your new JDBC data source.
* Indicates required fields

What would you like to name your new JDBC data source?

* Name: PatientDataSource

What JNDI name would you like to assign to your new JDBC Data Source?

JNDI Name: jdbc/PatientDataSource

What database type would you like to select?

Database Type: MySQL

Back | Next | Finish | Cancel

6. In the Name field, type `mpi_application_nameDataSource`.
The name you enter here will propagate elsewhere, and it must be the mpi application name that you have already created (for example, `PatientDataSource`).
7. In the JNDI Name field, type `jdbc/mpi_application_nameDataSource`.
Use the name you entered in step 6 here (for example, `jdbc/PatientDataSource`).
8. In the Database Type drop-down list, choose the appropriate type (for example: **MySQL**) and click **Next**.
9. In the Database Driver drop-down list, choose the appropriate driver (for example: **MySQL's Driver (Type 4) Versions:using com.mysql.jdbc.Driver**) and click **Next**.
10. Make sure that your data source supports Global Transactions.
 - At this step, for *MySQL only*, select **Emulate Two-Phase Commit**.
11. Click **Next**.

Figure 2–3 Create a New JDBC Data Source Panel - Connection Properties

The screenshot shows a dialog box titled "Create a New JDBC Data Source" with a "Connection Properties" tab selected. The dialog contains the following fields and values:

- Database Name:** Patient
- Host Name:** localhost
- Port:** 3306
- Database User Name:** mpi
- Password:** (masked with dots)
- Confirm Password:** (masked with dots)

12. In the Database Name field, type a name for the database to which you want to connect (for example: Patient).
13. In the Host Name field, type the name or the IP address of the database server (for example: localhost).
14. In the Port field, type the port on the database server that is used to connect to the database (for example: 3306).
15. In the Database User Name field, type the database account user name you want to use to create database connections (for example: patient).
16. In the Password field, type a password for your database account to use to create database connections.
17. In the Confirm Password field, re-type the password to confirm it.
18. Click **Next**.

The Settings for PatientDataSource page appears in the right panel.

19. Click the **Connection Pool** tab, click **Test Configuration**, and if successful click **Next**.

Note: If it fails, check the above steps.

Select Targets appears on the Create a New JDBC Data Source page in the right panel. Here you select one or more targets to deploy the new JDBC data source.

20. In the Servers check list, select one or more target servers and click **Finish**.
Deploy to the Clustered environment, if required.

Note: If you do not select a target, the data source will be created but not deployed. You will need to deploy the data source at a later time.

21. Repeat the above steps to create jdbc/PatientSequenceDataSource.
22. Repeat the above steps to create jdbc/PIXDomainLUDataSource. However, this time, make sure that you *do not* select **Support Global Transactions**, for it should *not* be checked.
23. Repeat the above steps to create jdbc/PIMPendingLinksDataSource.

To Create JDBC Data Resources for an IHE Profiles Application Project for Oracle

1. For instructions on how to start and stop your Oracle WebLogic Server, see *Starting and Stopping Servers: Quick Reference* at http://download.oracle.com/docs/cd/E14571_01/wls.htm.
2. Launch the **Oracle WebLogic Server Administration Console**.
3. Log in using the default user Name (**weblogic**) and Password (**welcome1**).
The Oracle WebLogic Administration Console appears.
4. On the left panel, under Domain Structure, expand **Services**, click **Data Sources**.
A Summary of JDBC Data Sources appears in the right panel.
5. To create a new JDBC Data Source click **New** at the bottom of the right panel and select **Generic Data Source** from the drop down list.
Settings for a new JDBC Data Source appears in the right panel of the page. It is here that you will create a new JDBC Data Source.
6. In the Name field, type `mpi_application_nameDataSource`.
The name you enter here will propagate elsewhere, and it must be the mpi application name that you have already created (for example, PatientDataSource).
7. In the JNDI Name field, type `jdbc/mpi_application_nameDataSource`.
Use the name you entered in step 6 here (for example, **jdbc/PatientDataSource**).
8. In the Database Type drop-down list, choose the appropriate type (for example: **Oracle**) and click **Next**.
9. In the Database Driver drop-down list, choose the appropriate driver; for example: **Oracle's Driver (Thin XA) for Instance Connections; Versions: 9.0.1 and later**.
10. Click **Next**.
Transaction Options appear on the page.
11. Click **Next**.
Connection Properties appears on the Create a New JDBC Data Source panel. Use it to define the connection properties.

12. In the Database Name field, type a name for the database to which you want to connect (for example: Patient).
13. In the Host Name field, type the name or the IP address of the database server (for example: localhost).
14. In the Port field, type the port on the database server that is used to connect to the database (for example: 1521).
15. In the Database User Name field, type the database account user name you want to use to create database connections (for example: patient).
16. In the Password field, type a password for your database account to use to create database connections.
17. In the Confirm Password field, re-type the password to confirm it.
18. Click **Next**.

The Settings for PatientDataSource page appears in the right panel.

19. Click **Test Configuration**, and if successful click **Next**.

Note: If it fails, check the above steps.

Select Targets appears on the Create a New JDBC Data Source page in the right panel. Here you select one or more targets to deploy the new JDBC data source.

20. In the Servers check list, select one or more target servers and click **Finish**.

Note: If you do not select a target, the data source will be created but not deployed. You will need to deploy the data source at a later time.

21. Repeat the above steps to create jdbc/PatientSequenceDataSource.
22. Repeat the above steps to create jdbc/PIXDomainLUDataSource.
23. Repeat the above steps to create jdbc/PIMPendingLinksDataSource.

Creating JMS Resources for an IHE Profile Application Project

JMS servers act as management containers for the queues and topics in the JMS modules that are targeted to them.

The following procedure includes instructions for creating JMS resources, which includes a:

- JMS Server
- JMS Module
- JMS Connection Factory in the specific JMS Module
- JMS Topic in the specific JMS Module

To Create JMS Server

1. On the left panel, under Domain Structure, expand **Services**, click **Messaging**, and then choose **JMS Servers**.

A Summary of JMS Servers appears in the right panel. It includes a table that summarizes the JMS servers that have been created in the current WebLogic Server domain.

Figure 2–4 Summary of JMS Servers Panel

Summary of JMS Servers

JMS servers act as management containers for the queues and topics in JMS modules that are targeted to them.

This page summarizes the JMS servers that have been created in the current WebLogic Server domain.

[▶ Customize this table](#)

JMS Servers (Filtered - More Columns Exist)

New Delete Showing 1 to 1 of 1 Previous | Next

<input type="checkbox"/>	Name ^	Persistent Store	Target	Current Server	Health
<input type="checkbox"/>	PatientJMServer		AdminServer	AdminServer	✔ OK

New Delete Showing 1 to 1 of 1 Previous | Next

2. In the table of previously created JMS Servers, click **New**.
The Create a New JMS Server panel appears.

Figure 2–5 Create a New JMS Server Panel

3. In the Name field, type the name for your new JMS Server.

Note: This name already exists in the table of previously created JMS Servers (in the example, **PatientJMSServer**).

4. Click **Next**.
Select targets appears in the right panel under Create a New JMS Server.
5. From the Target drop-down list, select a target server instance or migratable target on which you want to deploy the JMS Server.
If you are deploying on a Clustered environment, select the appropriate target server with the cluster environment.

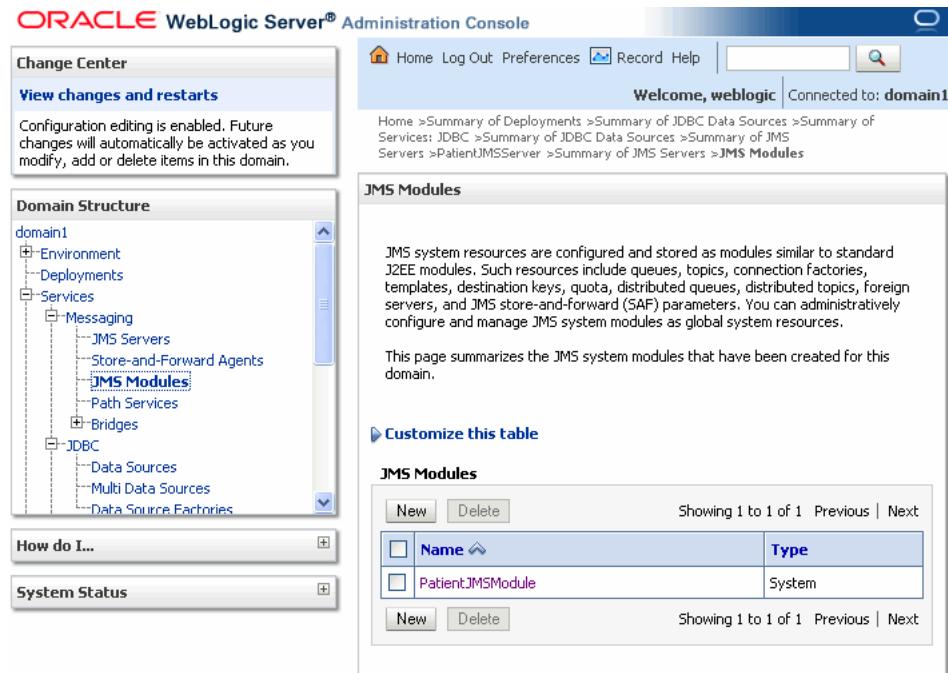
Note: The default server instance is exampleServer.

6. Click **Finish**.

To Create JMS Module

1. On the left panel, under Domain Structure, expand **Services**, click **Messaging**, and then choose **JMS Modules**.
The JMS Modules panel appears.

Figure 2–6 MS Modules Panel Summary



- In the JMS Modules table, click **New** to add a new JMS Module.
The Create JMS System Module panel appears.
- In the Name field, type the new JMS Module name (for example, *mpi_applicationNameJMSModule*).

Note: Again, remain consistent to the name chosen for the JDBC Data Source and the JMS Server (in the previous examples the key word was "Patient," making this name **PatientJMSModule**).

- Click **Next**.
Targets appears in the right panel under Create a New JMS System Module.
- In the Servers area, select the server or cluster on which you want to deploy this JMS system module.

Note: Retain the default, examplesServer.

- Click **Next**.
- Click **Finish**.

To Create JMS Connection Factory

- On the left panel, under Domain Structure, expand **Services**, click **Messaging**, and then choose **JMS Modules**.
- Choose the JMS Module (in the example, **PatientJMSModule**) from the table of JMS Modules.

The Settings for PatientJMSModule page appears in the right panel.

3. In the Summary of Resources table, click **New**.
4. Under the Type column in the Summary of Resources table choose **Connection Factory** and click **Next**.

Another panel of Create a New JMS System Module Resource appears.

5. In the Name field, type `PatientOutBoundSender`.
6. In the JNDI Name field, type `.jms/PatientOutBoundSender`.
7. Make sure that the **XA Connection Factory Enabled** check box is selected.
8. Click **Next**
9. In the Target field, retain the default server instance, which is `exampleServer`, and click **Finish**.

If you are deploying on a Clustered environment, select the appropriate target server with the cluster environment.

To Create JMS Topic

1. On the left panel, under Domain Structure, expand **Services**, click **Messaging**, and then choose **JMS Modules**.
2. In the right panel, choose the JMS Module you created (in the example, **PatientJMSModule**) from the table of JMS Modules.

Settings for PatientJMSModule appears in the right panel with a Summary of Resources table.

Figure 2–7 JMS Module Panel - Summary of Resources

The screenshot shows the 'Settings for PatientJMSModule' interface. It includes tabs for Configuration, Subdeployments, Targets, Security, and Notes. Below the tabs is a descriptive paragraph and a table of properties for the JMS module. At the bottom, there is a 'Summary of Resources' section with a table listing the resources created for this module.

Name	Type	JNDI Name	Subdeployment	Targets
PatientOutBoundSender	Connection Factory	./PatientOutBoundSender	Default Targetting	AdminServer
Patient.Topic	Topic	./Patient.Topic	Patient.Topic	PatientJMSModule
PixUpdateNotificationTopic	Topic	./PixUpdateNotificationTopic	PixUpdateNotificationTopic	PatientJMSModule

3. In the Summary of Resources table, click **New**, select **Topic**, and then click **Next**.
If you are deploying on a Clustered environment, select **DistributedTopic**.
The Create a New JMS System Module Resource panel appears on the right side of the window. Use this panel to set the properties that identify the new topic.
4. In the Name field, under JMS Destination Properties, type *mpi_application_nameTopic* (for example, *PatientTopic*).
5. Set **jms/PatientTopic** as the JNDI Name and click **Next**.
The Create a New JMS System Module Resource page appears in the right panel. Use this page to set the properties that will be used to target your new JMS system module resource.
6. In the Subdeployments drop-down list, select the topic name you just created (for example, *Patient Topic*) and click **Create a New Subdeployment**.
7. In the Subdeployment Name field, type *mpi_application_nameTopic* (for example, *PatientTopic*), and click **OK**.
8. In the Targets table of JMS Servers, select *mpi_application_nameJMS Server* (for example, **PatientJMS Server**).
9. Click **Finish**.

To Create **PixUpdateNotificationTopic**

1. In the Summary of Resources table (see [Figure 2-8](#)), click **New**, select **Topic**, and then click **Next**.
The Create a New JMS System Module Resource panel appears on the right side of the window. Use this panel to set the properties that identify the new topic.
2. In the Name field, under JMS Destination Properties, type *nameTopic* (for example, *PixUpdateNotificationTopic*).
3. Set **jms/PixUpdateNotificationTopic** as the JNDI Name and click **Next**.
The Create a New JMS System Module Resource page appears in the right panel. Use this page to set the properties that will be used to target your new JMS system module resource.
4. In the Subdeployments drop-down list, select the topic name you just created (for example, *PixUpdateNotification Topic*) and click **Create a New Subdeployment**.
5. In the Subdeployment Name field, type *nameTopic* (for example, *PixUpdateNotificationTopic*), and click **OK**.
6. In the Targets table of JMS Servers, select *nameJMS Server* (for example, **PatientJMS Server**).
7. Click **Finish**.

Creating JDBC Data Resources for an MPI Application Project

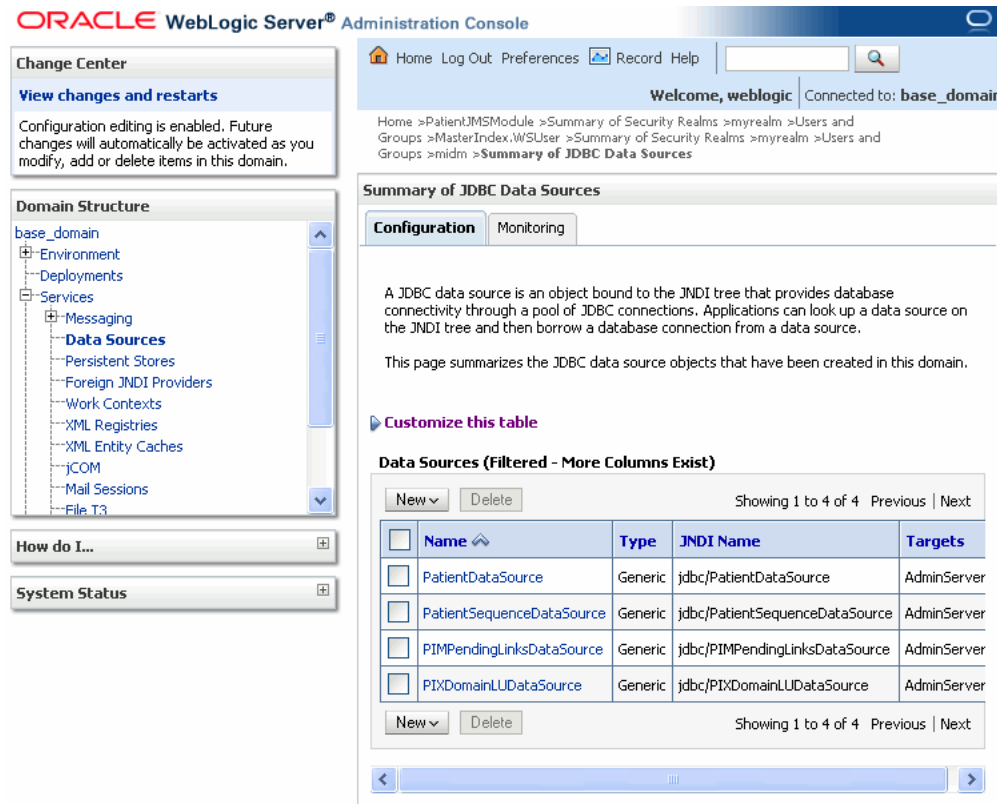
This section provides instructions for creating the JDBC data resources and defining the JDBC connections for an MPI Application Project.

To Create JDBC Data Resources for an MPI Application Project for MySQL

1. For instructions on how to start and stop your Oracle WebLogic Server, see *Starting and Stopping Servers: Quick Reference* at http://download.oracle.com/docs/cd/E14571_01/wls.htm.

2. Launch the **Oracle WebLogic Server Administration Console**.
3. Log in using the default user Name (**weblogic**) and Password (**welcome1**).
The Oracle WebLogic Administration Console appears.
4. On the left panel, under Domain Structure, expand **Services**, and then choose **Data Sources**.

Figure 2–8 WebLogic Administration Console - Summary of JDBC Data Sources



A Summary of JDBC Data Sources appears in the right panel.

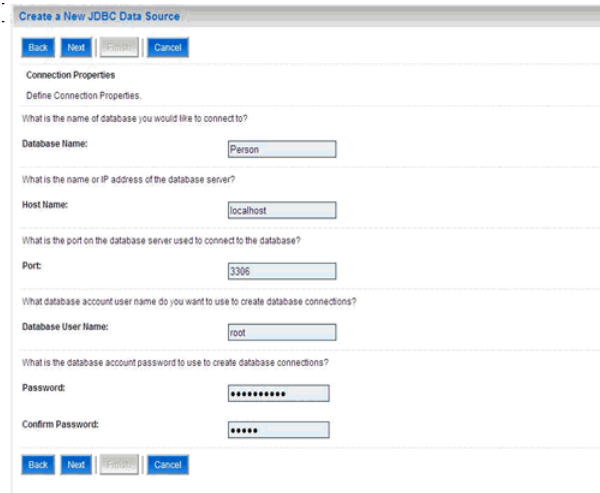
5. To create a new JDBC Data Source click **New** at the bottom of the right panel and select **Generic Data Source** from the drop down list.

Settings for a new JDBC Data Source appears in the right panel of the page. It is here that you will create a new JDBC Data Source.

Figure 2–9 Create a New JDBC Data Source Panel

6. In the Name field, type `mpi_application_nameDataSource`.
The name you enter here will propagate elsewhere, and it must be the mpi application name that you have already created (for example, `PersonDataSource`).
7. In the JNDI Name field, type `jdbc/mpi_application_nameDataSource`.
Use the name you entered in step 6 here (for example, `jdbc/PersonDataSource`).
8. In the Database Type drop-down list, choose the appropriate type (for example: **MySQL**) and click **Next**.
9. In the Database Driver drop-down list, choose the appropriate driver; for example: **MySQL's Driver (Type 4) Versions:using com.mysql.jdbc.Driver**, and click **Next**.
10. Make sure that your data source supports Global Transactions.
 - At this step, for MySQL *only*, select **Emulate Two-Phase Commit**.
11. Click **Next**.
Connection Properties appears on the Create a New JDBC Data Source panel. Use it to define the connection properties.

Figure 2–10 Create a New JDBC Data Source Panel - Connection Properties



12. In the Database Name field, type a name for the database to which you want to connect (for example: Person).
13. In the Host Name field, type the name or the IP address of the database server (for example: localhost).
14. In the Port field, type the port on the database server that is used to connect to the database (for example: 3306).
15. In the Database User Name field, type the database account user name you want to use to create database connections (for example: person).
16. In the Password field, type a password for your database account to use to create database connections.
17. In the Confirm Password field, re-type the password to confirm it.
18. Click **Next**.

The Settings for PersonDataSource page appears in the right panel.

19. Click **Test Configuration**, and if successful click **Next**.

Note: If it fails, check the above steps.

Select Targets appears on the Create a New JDBC Data Source page in the right panel. Here you select one or more targets to deploy the new JDBC data source.

20. In the Servers check list, select one or more target servers and click **Finish**.

If you are deploying on a Clustered environment, select the appropriate target server with the cluster environment.

Note: If you do not select a target, the data source will be created but not deployed. You will need to deploy the data source at a later time.

21. Repeat the above steps to create jdbc/PersonSequenceDataSource.

To Create JDBC Data Resources for an MPI Application Project for Oracle

1. For instructions on how to start and stop your Oracle WebLogic Server, see *Starting and Stopping Servers: Quick Reference* at http://download.oracle.com/docs/cd/E14571_01/wls.htm.
2. Launch the **Oracle WebLogic Server Administration Console**.
3. Log in using the default user Name (**weblogic**) and Password (**welcome1**).
The Oracle WebLogic Administration Console appears.
4. On the left panel, under Domain Structure, expand **Services**, click **JDBC**, and then choose **Data Sources**.
A Summary of JDBC Data Sources appears in the right panel.
5. To create a new JDBC Data Source click **New** at the bottom of the right panel and select **Generic Data Source** from the drop down list.
Settings for a new JDBC Data Source appears in the right panel of the page. It is here that you will create a new JDBC Data Source.
6. In the Name field, type *mpi_application_name*DataSource.
The name you enter here will propagate elsewhere, and it must be the mpi application name that you have already created (for example, PersonDataSource).
7. In the JNDI Name field, type *jdbc/object_name*DataSource.
Use the name you entered in step 6 here (for example, **jdbc/PersonDataSource**).
8. In the Database Type drop-down list, choose the appropriate type (for example: **Oracle**) and click **Next**.
9. In the Database Driver drop-down list, choose the appropriate driver; for example: **Oracle's Driver (Thin XA) for Instance Connections; Versions: 9.0.1 and later**.
10. Click **Next**.
Transaction Options appear on the page.
11. Click **Next**.
Connection Properties appears on the Create a New JDBC Data Source panel. Use it to define the connection properties.
12. In the Database Name field, type a name for the database to which you want to connect (for example: Person).
13. In the Host Name field, type the name or the IP address of the database server (for example: localhost).
14. In the Port field, type the port on the database server that is used to connect to the database (for example: 1521).
15. In the Database User Name field, type the database account user name you want to use to create database connections (for example: person).
16. In the Password field, type a password for your database account to use to create database connections.
17. In the Confirm Password field, re-type the password to confirm it.
18. Click **Next**.
The Settings for PersonDataSource page appears in the right panel.
19. Click **Test Configuration**, and if successful click **Next**.

Note: If it fails, check the above steps.

Select Targets appears on the Create a New JDBC Data Source page in the right panel. Here you select one or more targets to deploy the new JDBC data source.

20. In the Servers check list, select one or more target servers and click **Finish**.

If you are deploying on a Clustered environment, select the appropriate target server with the cluster environment.

Note: If you do not select a target, the data source will be created but not deployed. You will need to deploy the data source at a later time.

21. Repeat the above steps to create jdbc/PersonSequenceDataSource.

Creating JMS Resources for an MPI Application Project

JMS servers act as management containers for the queues and topics in the JMS modules that are targeted to them.

The following procedure includes instructions for creating JMS resources, which includes a:

- JMS Server
- JMS Module
- JMS Connection Factory in the specific JMS Module
- JMS Topic in the specific JMS Module

To Create JMS Server

1. On the left panel, under Domain Structure, expand **Services**, click **Messaging**, and then choose **JMS Servers**.

A Summary of JMS Servers appears in the right panel. It includes a table that summarizes the JMS servers that have been created in the current WebLogic Server domain.

Figure 2–11 Summary of JMS Servers Panel

Summary of JMS Servers

JMS servers act as management containers for the queues and topics in JMS modules that are targeted to them.

This page summarizes the JMS servers that have been created in the current WebLogic Server domain.

[Customize this table](#)

JMS Servers (Filtered - More Columns Exist)

Showing 1 to 4 of 4 Previous | Next

<input type="checkbox"/>	Name ↕	Persistent Store	Target	Current Server	Health
<input type="checkbox"/>	CompanyJMSServer		examplesServer	examplesServer	✔ OK
<input type="checkbox"/>	examplesJMSServer	exampleJDBCStore	examplesServer	examplesServer	✔ OK
<input type="checkbox"/>	PersonJMSServer		examplesServer	examplesServer	✔ OK
<input type="checkbox"/>	WseeJMSServer	WseeFileStore	examplesServer	examplesServer	✔ OK

Showing 1 to 4 of 4 Previous | Next

- In the table of previously created JMS Servers, click **New**.
The Create a New JMS Server panel appears.

Figure 2–12 Create a New JMS Server Panel

Create a New JMS Server

JMS Server Properties

The following properties will be used to identify your new JMS Server.

* Indicates required fields

What would you like to name your new JMS Server?

* Name:

Specify persistent store for the new JMS Server.

Persistent Store:

- In the Name field, type the name for your new JMS Server.

Note: This name already exists in the table of previously created JMS Servers (in the example, **PersonJMSServer**).

- Click **Next**.
Select targets appears in the right panel under Create a New JMS Server.
- From the Target drop-down list select a target server instance or migratable target on which you want to deploy the JMS Server.

If you are deploying on a Clustered environment, select the appropriate target server with the cluster environment.

Note: The default server instance is exampleServer.

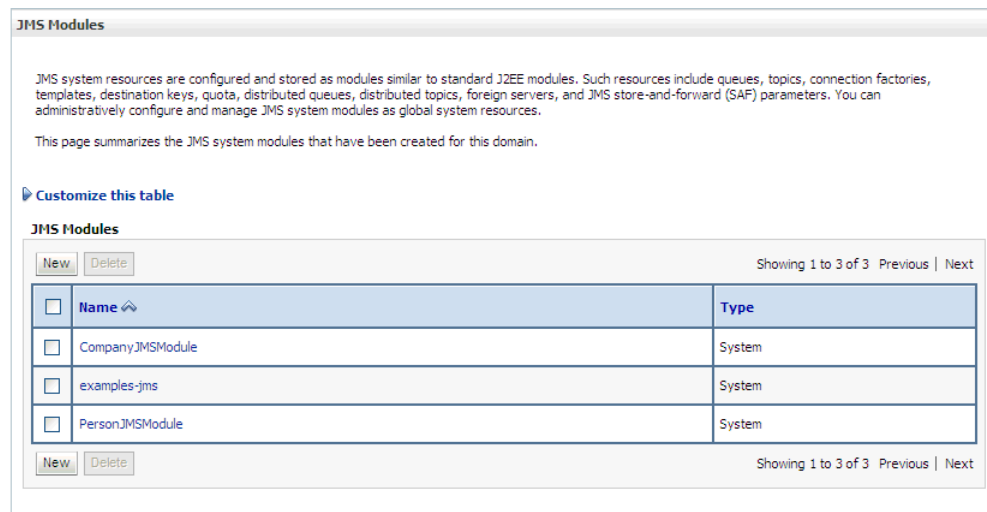
6. Click **Finish**.

To Create JMS Module

1. On the left panel, under Domain Structure, expand **Services**, click **Messaging**, and then choose **JMS Modules**.

The JMS Modules panel appears.

Figure 2–13 JMS Modules Panel - JMS Modules Table



2. In the JMS Modules table, click **New** to add a new JMS Module.
The Create JMS System Module panel appears.
3. In the Name field, type the new JMS Module name (for example, *mpi_applicationNameJMSModule*).

Note: Again, remain consistent to the name chosen for the JDBC Data Source and the JMS Server (in the previous examples the key word was "Person," making this name **PersonJMSModule**).

4. Click **Next**.
Targets appears in the right panel under Create a New JMS System Module.
5. In the Servers area, select the server or cluster on which you want to deploy this JMS system module.

If you are deploying on a Clustered environment, select the appropriate target server with the cluster environment.

Note: Retain the default, `examplesServer`.

6. Click **Finish**.

To Create JMS Connection Factory

1. On the left panel, under Domain Structure, expand **Services**, click **Messaging**, and then choose **JMS Modules**.
2. Choose the JMS Module (in the example, **PersonJMSModule**) from the table of JMS Modules.

The Settings for PersonJMSModule page appears in the right panel.

3. In the Summary of Resources table, click **New**.
4. Under the Type column in the Summary of Resources table choose **Connection Factory** and click **Next**.

Another panel of Create a New JMS System Module Resource appears.

5. In the Name field, type `PersonOutBoundSender`.
6. In the JNDI Name field, type `jms/mpi_application_nameOutBoundSender`.
7. Make sure that the **XA Connection Factory Enabled** check box is selected.
8. Click **Next**
9. In the Target field, retain the default server instance, which is `exampleServer`, and click **Finish**.

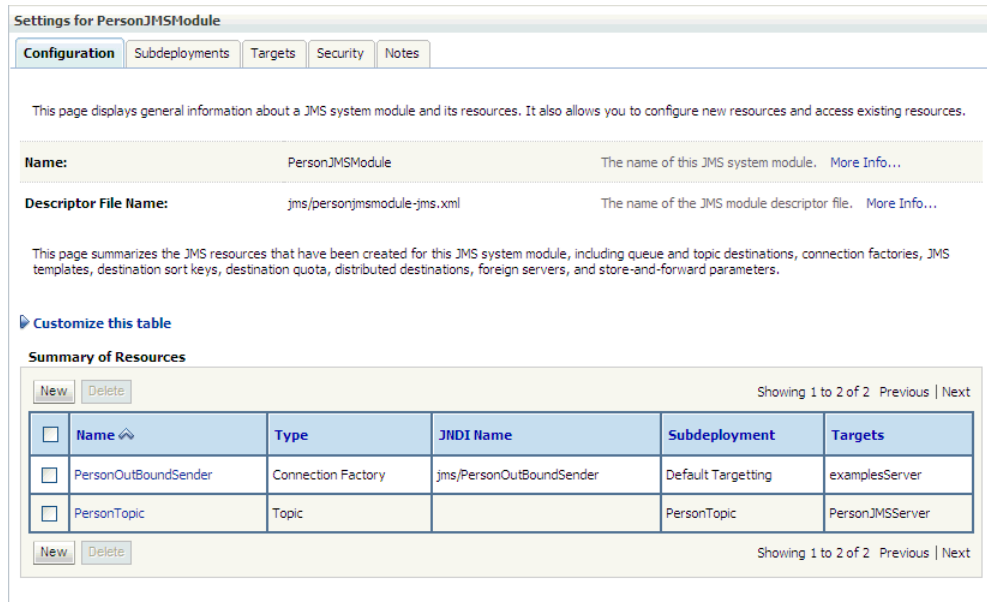
If you are deploying on a Clustered environment, select the appropriate target server with the cluster environment.

To Create JMS Topic

1. On the left panel, under Domain Structure, expand **Services**, click **Messaging**, and then choose **JMS Modules**.
2. In the right panel, choose the JMS Module you created (in the example, **PersonJMSModule**) from the table of JMS Modules.

Settings for PersonJMSModule appears in the right panel with a Summary of Resources table.

Figure 2–14 Settings for PersonJMSModule Panel - Summary of Resources Table



3. In the Summary of Resources table, click **New**, select **Topic**, and then click **Next**.
The Create a New JMS System Module Resource panel appears on the right side of the window. Use this panel to set the properties that identify the new topic.
4. In the Name field, under JMS Destination Properties, type *mpi_application_nameTopic* (for example, *PersonTopic*).
5. Set *jms/PersonTopic* as the JNDI Name and click **Next**.
The Create a New JMS System Module Resource page appears in the right panel. Use this page to set the properties that will be used to target your new JMS system module resource.
6. In the Subdeployments drop-down list, select the topic name you just created (for example, *PersonTopic*) and click **Create a New Subdeployment**.
7. In the Subdeployment Name field, type *mpi_application_nameTopic* (for example, *PersonTopic*), and click **OK**.
8. In the Targets table of JMS Servers, select *mpi_application_nameJMSServer* (for example, **PersonJMSServer**).
If you are deploying on a Clustered environment, select the appropriate target server with the cluster environment.
9. Click **Finish**.

Setting Up the User

In this step you create the MasterIndex.Admin and Administrator groups, and then create a new user within the two groups.

1. On the left panel, under Domain Structure, expand **Services**, and then choose **Security Realms**.
2. In the table on the Summary of Security Realms panel, click **myrealm**, which is the name of the realm.

The Settings for myrealm panel appears.

Figure 2–15 Settings for myrealm Panel

3. Select the **Users and Groups** tab and then click **Groups**.
4. In the Groups table, click **New**.
5. In the Name field, type `MasterIndex.Admin` and click **OK**.
6. In the Groups table, click **New**.
7. In the Name field, type `Administrator` and click **OK**.
8. On the Settings for myrealm panel, select **Users and Groups** and then **Users**.
9. In the Users table, click **New**.
10. Type a name and a password for the new user you are creating and click **OK**.
11. Select **User Group**.
12. To add the two groups you created to the user you created, from the Available list, drag `MasterIndex.Admin` to the **Chosen** list, and then drag `Administrator` to the **Chosen** list.
13. Add the user `MasterIndex.WSUser` and then add the `MasterIndex.Admin` group for it.

Note: See *Oracle Healthcare Master Person Index User's Guide*.

Using MPI and IHE Profile Applications on WebLogic

This chapter provides procedures that explain how to deploy and run an MPI Application or an IHE Profile Application on an Oracle WebLogic Server.

This chapter includes the following section:

- [Deploying and Running Applications on Oracle WebLogic Server](#) on page 1
- [Web Layer Separation for OHMPI](#) on page 3
- [WebLogic Clustering](#) on page 4

Deploying and Running Applications on Oracle WebLogic Server

This procedure leads you through the steps to deploy and run an IHE Profiles Application and an MPI Application on Oracle WebLogic Server.

To Deploy and Run Applications on an Oracle WebLogic Server

The procedure is the same for an IHE Profiles Application and an MPI Application, except step 7, and the differences are pointed out.

1. On the left panel of the WebLogic Server Administration Console, under Domain Structure, select **Environment** and then choose **Deployments**.

The Summary of Deployments panel appears.

2. On the right side of the panel under Deployments, click **Install**.

A Summary of Deployments panel with a Deployments table containing a list of EAR files appears.

Figure 3–1 Summary of Deployments Panel - Deployments Table

Summary of Deployments

Control Monitoring

This page displays a list of Java EE applications and stand-alone application modules that have been installed to this domain. Installed applications and modules can be started, stopped, updated (redeployed), or deleted from the domain by first selecting the application name and using the controls on this page.

To install a new application or module for deployment to targets in this domain, click the Install button.

[Customize this table](#)

Deployments

Install Update Delete Start Stop Showing 1 to 10 of 15 Previous Next

<input type="checkbox"/>	Name	State	Health	Type	Deployment Order
<input type="checkbox"/>	apache_xbean.jar	Active		Library	100
<input type="checkbox"/>	asyncServletEar	Active	OK	Enterprise Application	100
<input type="checkbox"/>	ejb20BeanMgedEar	Active	OK	Enterprise Application	100
<input type="checkbox"/>	ejb30	Active	OK	Enterprise Application	100
<input type="checkbox"/>	examplesWebApp	Active	OK	Web Application	100
<input type="checkbox"/>	extServletAnnotationsEar	Active	OK	Enterprise Application	100
<input type="checkbox"/>	jdbcRowSetsEar	Active	OK	Enterprise Application	100
<input type="checkbox"/>	jspSimpleTagEar	Active	OK	Enterprise Application	100
<input type="checkbox"/>	mainWebApp	Active	OK	Web Application	100
<input type="checkbox"/>	pubsub(1.0, 1.5.0.0)	Active		Library	100

Install Update Delete Start Stop Showing 1 to 10 of 15 Previous Next

3. Locate your application EAR and click **Next** (in the procedures in [Chapter 2, "Oracle WebLogic Server Configuration."](#))

The Install Application Assistant page appears in the right panel.

4. Locate the deployment you want to install and prepare for deployment.

Tip: Select the file path that represent the application root directory, archive file, exploded archive directory, or application module descriptor that you want to install. You can also enter the path of the application directory or file in the Path field.

Note: Only valid file paths are displayed. If you cannot find your deployment files, upload your file(s) and/or confirm that your application contains the required deployment descriptors.

5. Click **Next**.

Note: When deploying an MPI EAR file through the WebLogic Admin Console, under Security make sure that you choose **DD Only**. If you choose one of the other options, you will not be able to log into the MIDM.

6. Click **Finish**.

7. Launch **Master Index Data Manager** (MIDM).
8. From a web browser, enter the following:
 - For IHE Profiles Application: <http://localhost:7001/PatientMIDM>
 - For MPI Application: <http://localhost:7001/PersonMIDM>

Note: The "Patient" portion of PatientMIDM or the "Person" portion of PersonMIDM is from the object name that you previously set. "Patient" is mandatory for an IHE Profiles while "Person" could be another object name.

9. Log in using your user name and password.

Web Layer Separation for OHMPI

To separate the Web tier from the Business Logic tier, the MIDM UI layer is separated from the MIDM EJB layer. The MIDM UI layer and the MIDM EJB layer that are currently packaged into one EAR deployment package are broken down into individual WAR and EJB deployment packages.

Standalone WAR and EJB deployment packages are generated and these are setup on two different servers. The default EAR is also generated and is used for the consolidated deployment. The `ohmpi-midm-connector.properties` file is modified for setting up the WAR deployment to communicate with the EJB deployment.

To separate the web layer from the EJB layer:

1. [Building the Project](#) on page 3
2. [Configuring the Application Server](#) on page 3
3. [Configuring Connectivity](#) on page 4
4. [Deploying the Build](#) on page 4

Building the Project

- For a new project, two additional deployment packages (WAR and EJB) are available in the dist folder after you build the project.
 - WebLogic, `<APP-NAME>-war.war` and `<APP-NAME>-ejb.ear`
- For existing projects, execute the Clean Project and Generate Master Index files to update the project before building it.

Configuring the Application Server

To configure the servers hosting the individual deployment packages:

1. Define users with the same username, password, and roles on both the servers.
2. Define the JDBC Data Sources (`<APP-NAME>DataSource` and `<APP-NAME>SequenceDataSource`) on the server deploying the EJB component.
3. Define the JMS Resource, `<APP-NAME>OutBoundSender` and `<APP-NAME>Topic`, on the server deploying the EJB component as well as on the server deploying the WAR component.

Configuring Connectivity

1. Create an `ohmpi-midm-connector.properties` file that defines the EJB connectivity information:
 - `jndi-initial-context-factory-name=weblogic.jndi.WLInitialContextFactory`
 - `ohmpi-ejb-url=<remote-weblogic-provider-url>` (For example: `iiop://127.0.0.1:7001` or `t3://127.0.0.1:7001`)
 - `username=<USERNAME defined during user setup>`
 - `password=<PASSWORD given during user setup>`
2. Drop this property file into `<app-server-domain-config-directory>/ohmpi/` on the server for Web Layer deployment. For example:
 - WebLogic:
`user-projects/domains/<domain-name>/config/ohmpi/ohmpi-midm-connector.properties`

Deploying the Build

Deploy the EJB and WAR package on the EJB Server and Web Server, respectively.

Note: The cross-server deployment is not supported. You must deploy both the UI and EJB deployment packages on the WebLogic Servers.

WebLogic Clustering

To deploy MPI to a Clustered environment:

1. Create two clusters if you are using the web layer isolated environment.
2. Configure the nodes as the Web servers are configured under one cluster, such as `MIDM_Web_Cluster`.
3. Configure the nodes as the EJB servers are configured under one cluster, such as `MIDM_EJB_Cluster`.
4. Deploy the JMS Resources on both the `MIDM_Web_Cluster` and `MIDM_EJB_Cluster`.

Ensure that `DistributeTopic` is selected while creating the Topic JMS resource. See [To Create JMS Topic](#) on page 12.
5. Deploy the JDBC Resources on `MIDM_EJB_Cluster`.
6. Create the same username and password combination for users on the nodes for the WEB and EJB clusters within the security realm.
7. Deploy the `Person-war.war` on `MIDM_Web_Cluster`.
8. Deploy the `Person-ejb.ear` on `MIDM_EJB_Cluster`.
9. Set up a ProxyWebApp Web application on the Admin Server. To do so:
 - a. Create a ProxyWebApp Web Application in NetBeans for the WebLogic server with the folder structure as shown in [Figure 3-2](#).

Figure 3–2 Folder Structure**b. Modify the index.jsp file:**

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <hi>Hello World!</h1>
  </body>
</html>
  
```

c. Modify the weblogic.xml file:

```

<?xml version="1.0" encoding="UTF-8"?>
<weblogic-web-app xmlns="http://www.bea.com/ns/weblogic/90">
  <context-root></context-root>
</weblogic-web-app>
  
```

d. Modify the web.xml file:

```

param-name as WebLogicCluster
param-value as <Machine-A IP Address>:7003|<Machine-B IP Address>:7003
  
```

The parameters includes all servers on which the web application are deployed.

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd">
  <servlet>
    <servlet-name>HttpClusterServlet</servlet-name>
    <servlet-class>weblogic.servlet.proxy.HttpClusterServlet</servlet-class>
    <init-param>
      <param-name>WebLogicCluster</param-name>
      <param-value>10.178.187.164:7003|10.178.187.132:7003</param-value>
    </init-param>
    <init-param>
      <param-name>verbose</param-name>
      <param-value>true</param-value>
    </init-param>
    <init-param>
      <param-name>DebugConfigInfo</param-name>
      <param-value>ON</param-value>
    </init-param>
  </servlet>
  <servlet-mapping>
    <servlet-name>HttpClusterServlet</servlet-name>
    <url-pattern>/</url-pattern>
  </servlet-mapping>
  
```

```
<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.jsp</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.jspx</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.jsf</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.htm</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.html</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.do</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.jpg</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.gif</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.css</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.js</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.png</url-pattern>
</servlet-mapping>

<servlet-mapping>
  <servlet-name>HttpClusterServlet</servlet-name>
  <url-pattern>*.action</url-pattern>
</servlet-mapping>
```

```
</web-app>
```

10. Deploy the ProxyWebApp Web application on the system acting as the LoadBalancer system.
11. For EJB replication and fail-over, configure multiple EJB deployments in ohmpi-midm-connector.properties by setting the ohmpi-ejb-url as below:
ohmpi-ejb-url=t3://<ejb-server-1-IP>:<Port>,<ejb-server-2-IP>:Port
where ejb-server-1 and ejb-server-2 are the servers on which the EJB is deployed.
12. In order to access the load balanced, replicated and the fail-over enabled MIDM application, access the MIDM application through the proxywebapp application.

If the ProxyWebApp Web application is deployed on the LoadBalancer server running on port 7001, then access the MIDM application using:

```
http://< LoadBalancer _IP_Address>:7001/<APP-NAME>MIDM/login.jsf
```

