

Oracle® Communications
MetaSolv Solution
LSR 9.4
Installation Guide

April 2011

Oracle Communications MetaSolv Solution LSR 9.4 Installation Guide.

Copyright © 2011, 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 software or related documentation 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 USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software 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 which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software 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
MetaSolv Solution LSR Documentation Set	v
1. Installation and Deployment	1
Pre-Installation Checklist	1
Installing LSR 9.4	1
Applying LSR Database Changes	3
Deploying LSR on the Application Server in a Single Server Environment	3
Deploying LSR on the Application Server in a Clustered Environment	6
If You Are Using LSR with XML APIs	9
Creating Additional Queues on a Single Server	9
Creating Additional Queues on a Clustered Server	11
Deploying LSR Integration on the Application Server in a Single Server Environment	15
Deploying LSR Integration on the Application Server in a Clustered Environment	19
About the LSR 9 Samples File	21

Preface

This guide contains the procedures and information you need to install LSR 9.4 and future service packs and EFixes of the LSR 9.4 release stream. It provides instructions for installing on Oracle WebLogic 10.3.1. Follow the instructions for the application server that you use.

For future service packs, refer to the respective LSR release notes document for the minimum required version of MetaSolv Solution. You should always install different LSR versions in sequence, starting with the earlier LSR version.

This guide includes some information on third-party software products used by MetaSolv Solution. However, this is limited to information needed to install and perform initial configuration tasks. If you need additional information on a third-party software application, consult the documentation provided by the product's manufacturer.

This guide contains the procedures and information you need to install and initially configure MetaSolv Solution and its utility programs.

This guide includes some information on third-party software products used by MetaSolv Solution. However, this information is limited to information needed to install and perform initial configuration tasks. If you need additional information on a third-party software application, consult the documentation provided by the product's manufacturer.

Audience

This guide is for individuals responsible for installing or maintaining MetaSolv Solution and ensuring the software is operating as required. This guide assumes the installer has an Oracle DBA and WebLogic administrator background, with a working knowledge of Windows 2000 and higher and JEE.

MetaSolv Solution LSR Documentation Set

In addition to this guide, Oracle has published the *LSR Developer's Reference* to provide specific information about the LSR API. For information on installing the MetaSolv Solution core application, refer to the various documentation located on E-Delivery.

Installation and Deployment

This chapter explains how to install LSR 9.4 on Oracle WebLogic Server 10.3.1 for all platforms.

Pre-Installation Checklist

Ensure the following:

- ◆ You have installed Oracle Communications MetaSolv Solution 6.2.
- ◆ The following environment variables are set:

```
PATH= BEA_Home/jdk_version/bin $PATH
BEA_HOME=/opt/m62Single/single/bea
JAVA_HOME=/opt/m62Single/single/bea/jdk_version
```

where:

BEA_Home is the directory in which Oracle WebLogic Server 10.3.1 is installed.

Installing LSR 9.4

To install LSR 9.4:

1. From *MSLV_Home*, enter the following command:

```
java -jar Lsr.releaseNo.buildNo_M62.jar
```

where:

MSLV_Home is the directory in which the MetaSolv Solution software is installed.

releaseNo is the release version of LSR.

buildNo is the build number that contains the LSR installer.

For example:

```
java -jar Lsr.R9_4_0.b43_M62.jar
```

The Select MetaSolv home directory window appears.

2. From the **Look In** list, select a directory in which the installation program files can be stored and used during the installation process, and then click **Select**.

A directory named **lsr9.4installer** is automatically created in the selected directory. Oracle recommends choosing your *MSLV_Home* directory for this task.

The installation program launches the installation process by running **setup.sh** (UNIX/Linux) or **setup.cmd** (Windows) in the *MSLV_Home/lsr9.4installer* directory.

The welcome screen for the MetaSolv Solution LSR installation appears.

3. Click **Next**.

The Details window appears.

4. Enter the path and name of the MetaSolv Home directory, MetaSolv Domain directory, and MetaSolv Server directory or click **Open** to search for the directories.
5. Click **Next**.

The Install Type window appears.

6. Select one of the following installation types:
 - ◆ Install all LSR Files
 - ◆ Install Only LSR EAR File

When prompted for the Install Type, if the environment is a clustered environment, you can choose an option to extract the **.EAR** file only. Use this option to extract the **.EAR** file on the Admin server for deployment to separate managed servers. For a non-clustered environment, choose the option to install and deploy all files.

7. Click **Next**.

The Install Summary window appears.

8. Click **Finish**.

9. (Optional) If you are installing LSR in a clustered environment, rename the **loggingconfig_LSR9.xml** file to **loggingconfig_cluster-LSR9.xml**.

The **loggingconfig_LSR9.xml** file is located in the *MSLV_Home/managedserver/appserver/config* directory, where *managedserver* is the WebLogic managed server.

Applying LSR Database Changes

You must apply the contents of the **prodfixsql** directory.

To apply LSR database changes:

1. Run the **lsr_master.sql** file located in the *MSLV_Home/server/appserver/sql/LSR9/prodfixsql* directory, where *server* is the name of the WebLogic server.
2. Run **DBHealth**, which recompiles all invalid objects.

Deploying LSR on the Application Server in a Single Server Environment

Deploying LSR on the application server in a single-server environment involves the following:

- ◆ Undeploying the existing LSR 9 application from the application server
- ◆ Deploying LSR 9.4 on the application server

To undeploy LSR 9 from the application server:

1. Start the WebLogic Server Administration Console using the following URL:

`http://ServerName:Port/console`

where:

ServerName is the name of the administration server.

Port is the administration server port number.

For example:

`http://wplsunsvxmlapi:7001/console`

2. Enter the administration user name and password when prompted and press **Enter**.
3. Under **Change Center**, click **Lock & Edit**.
4. Expand the **Domain Structure** tree and click **Deployments**.
The Summary of Deployments window appears.
5. On the **Control** tab, select **LSR9**.
6. From the **Stop** list, select **Force Stop Now**.
Ensure that the state of the LSR 9 application has changed from **Active** to **Prepared**.
7. Select **LSR9** and click **Delete**.
The Delete Application Assistant window appears.
8. Click **Yes**.

The LSR 9 application is undeployed.

Before you deploy LSR 9.4, ensure the following servers are running:

◆ Administration server

If it is not running, start it using the following startup script:

For UNIX:

```
domain_directory/startServerName.sh
```

For Windows:

```
domain_directory/startServerName.cmd
```

where:

domain_directory is the WebLogic server domain directory.

ServerName is the name of the administration server.

To deploy LSR 9.4 on the application server:

1. Start the WebLogic Server Administration Console using the following URL:

```
http://ServerName:Port/console
```

where:

ServerName is the name of the administration server.

Port is the administration server port number.

For example:

```
http://wplsunsvxmlapi:7001/console
```

2. Enter the administration user name and password when prompted and press **Enter**.
3. Under **Change Center**, click **Lock & Edit**.
4. Expand the **Domain Structure** tree and click **Deployments**.
The Summary of Deployments window appears.
5. On the **Control** tab, click **Install**.
The Install Application Assistant window appears.
6. Under **Current Location**, navigate to the directory from which you want to deploy LSR 9.4 (for example, **opt/metasolv/mslv01/appserver/deploy**), select **LSR9.ear**, and then click **Next**.
The Choose targeting style window appears.
7. Select **Install this deployment as an application** and click **Next**.

8. Under **Source accessibility**, select **I will make the deployment accessible from the following location**.
9. Click **Finish**.
10. Under **Change Center**, click **Activate Changes**.
11. Expand the **Domain Structure** tree and click **Deployments**.

The Summary of Deployments window appears.

12. Select **LSR9**.
13. From the **Start** list, select **Servicing all requests**.

The Start Application Assistant window appears.

14. Click **Yes**.

Ensure that the state of the LSR 9 application has changed from **Prepared** to **Active**.

15. If you receive the following error in the **appserver.mss.log** file during LSR deployment,

```
"log4j: ERROR Attempted to append to closed appender named
[XMLFileApp"
```

do the following:

- a. Open the **loggingconfig.xml** file located in the *MSLV_Home/server/appserver/config* directory, where *server* is the name of the WebLogic server.
- b. Remove the following entries:

```
category name="cmm.LSR"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLogger"
additivity="false"><level value ="error"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLevel" />
<appender-ref ref="XMLFileApp" />
</category>
<category name="LSR"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLogger"
additivity="false">
<level value="error"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLevel" />
<appender-ref ref="XMLFileApp" />
</category>
<category name="cmm.LSR"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLogger"
additivity="false">
<level value ="error"
```

```
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLevel"/>
<appender-ref ref="XMLFileApp"/>
</category>
```

- c. Save and close the file.

Deploying LSR on the Application Server in a Clustered Environment

If the administration server for the domain is on a machine where no managed server process is running, you can run the LSR installer and select the option to only extract the **.ear** file. The **.ear** file must reside on the machine where the admin server is running to deploy to the managed servers in the domain.

Deploying LSR on the application server in a clustered-server environment involves the following:

- ◆ Undeploying the existing LSR 9 application from the application server
- ◆ Deploying LSR 9.4 on the application server

To undeploy LSR 9 from the application server:

1. Start the WebLogic Server Administration Console using the following URL:

`http://ServerName:Port/console`

where:

ServerName is the name of the administration server.

Port is the administration server port number.

For example:

`http://wplsunsvxmlapi:7001/console`

2. Enter the administration user name and password when prompted and press **Enter**.
3. Under **Change Center**, click **Lock & Edit**.
4. Expand the **Domain Structure** tree and click **Deployments**.
The Summary of Deployments window appears.
5. On the **Control** tab, select **cluster-LSR9**.
6. From the **Stop** list, select **Force Stop Now**.
Ensure that the state of the LSR 9 application has changed from **Active** to **Prepared**.
7. Select **cluster-LSR9** and click **Delete**.
The Delete Application Assistant window appears.
8. Click **Yes**.

The LSR 9 application is undeployed.

Before you deploy LSR 9.4 deployment, ensure the following servers are running:

◆ Administration server

If it is not running, start it using the following startup script:

For UNIX:

domain_directory/**startAdminServer.sh** (for non-XML deployment)

or

domain_directory/**startAdminServer_wli.sh** (for XML deployment)

For Windows:

domain_directory/**startAdminServer.cmd** (for non-XML deployment)

or

domain_directory/**startAdminServer_wli.cmd** (for XML deployment)

where:

domain_directory is the WebLogic server domain directory.

◆ Managed server or servers

Start any managed servers that are not running using the following startup script:

For UNIX:

domain_directory/**startServerName.sh**

For Windows:

domain_directory/**startServerName.cmd**

where:

domain_directory is the WebLogic server domain directory.

ServerName is the name of the administration server.

To deploy LSR 9.4 on the application server:

1. Start the WebLogic Server Administration Console using the following URL:

<http://ServerName:Port/console>

where:

ServerName is the name of the administration server.

Port is the administration server port number.

For example:

http://wplsunsvxmlapi:7001/console

2. Enter the administration user name and password when prompted and press **Enter**.

3. Under **Change Center**, click **Lock & Edit**.

4. Expand the **Domain Structure** tree and click **Deployments**.

The Summary of Deployments window appears.

5. On the **Control** tab, click **Install**.

The Install Application Assistant window appears.

6. Under **Current Location**, navigate to the directory from which you want to deploy LSR 9.4 (for example, **opt/metasolv/mslv01/appserver/deploy**), select **cluster-LSR9.ear**, and then click **Next**.

The Choose targeting style window appears.

7. Select **Install this deployment as an application** and click **Next**.

The Select deployment targets window appears.

8. Under **Clusters**, select **All servers in the cluster**.

9. Under **Source accessibility**, select **I will make the deployment accessible from the following location**.

10. Click **Finish**.

11. Under **Change Center**, click **Activate Changes**.

12. Expand the **Domain Structure** tree and click **Deployments**.

The Summary of Deployments window appears.

13. Select **cluster-LSR9**.

14. From the **Start** list, select **Servicing all requests**.

The Start Application Assistant window appears.

15. Click **Yes**.

Ensure that the state of the LSR 9 application has changed from **Prepared** to **Active**.

16. If you receive the following error in the **appserver.mss.log** file during LSR deployment,

```
"log4j: ERROR Attempted to append to closed appender named  
[XMLFileApp"
```

do the following:

- a. Open the **loggingconfig.xml** file located in the *MSLV_Home/managedserver/***appserver/config** directory, where *managedserver* is the WebLogic managed server.

- a. Remove the following entries:

```

category name="cmm.LSR"

class="com.metasolv.common.framework.logging.api.log4jext.MSLVLogger"
additivity="false"><level value ="error"

class="com.metasolv.common.framework.logging.api.log4jext.MSLVLevel" />
<appender-ref ref="XMLFileApp" />
</category>

<category name="LSR"

class="com.metasolv.common.framework.logging.api.log4jext.MSLVLogger"
additivity="false">

<level value="error"

class="com.metasolv.common.framework.logging.api.log4jext.MSLVLevel" />
<appender-ref ref="XMLFileApp" />
</category>

<category name="cmm.LSR"

class="com.metasolv.common.framework.logging.api.log4jext.MSLVLogger"
additivity="false">

<level value ="error"

class="com.metasolv.common.framework.logging.api.log4jext.MSLVLevel" />
<appender-ref ref="XMLFileApp" />
</category>

```

- b. Save and close the file.

If You Are Using LSR with XML APIs

If you are using LSR and MetaSolv Solution with the XML API option, you must create the following additional queues on the WebLogic server:

- ◆ LSR9_API.queue.AsyncDispatcher
- ◆ LSR9_API.queue.AsyncDispatcher_error

Creating Additional Queues on a Single Server

To create additional queues on a single server:

1. Create the **LSR9_API.queue.AsyncDispatcher** queue:
 - a. Log on to the WebLogic Server Administration Console.
 - b. In the **Domain Structure** tree, expand **Services**, expand **Messaging**, and then select **JMS Modules**.
The JMS Modules page appears.
 - c. Click **conversational-jms**.

- The Settings for conversational-jms page appears.
- d. On the **Configuration** tab, under **Summary of Resources**, click **New**.
The Create a New JMS System Module Resource page appears.
 - e. Select **Queue** and click **Next**.
 - f. In the **Name** field, enter **LSR9_API.queue.AsyncDispatcher**.
 - g. In the **JNDI Name** field, enter **LSR9_API.queue.AsyncDispatcher**.
The queue name and JNDI name must be the same.
 - h. Click **Next**.
 - i. From the **Subdeployments** list, select **cgJMSServer**.
 - j. Click **Finish**.
2. Create the **LSR9_API.queue.AsyncDispatcher_error** queue:
- a. Log on to the WebLogic Server Administration Console.
 - b. In the **Domain Structure** tree, expand **Services**, expand **Messaging**, and then select **JMS Modules**.
The JMS Modules page appears.
 - c. Click **conversational-jms**.
The Settings for conversational-jms page appears.
 - d. On the **Configuration** tab, under **Summary of Resources**, click **New**.
The Create a New JMS System Module Resource page appears.
 - e. Select **Queue** and click **Next**.
 - f. In the **Name** field, enter **LSR9_API.queue.AsyncDispatcher_error**.
 - g. In the **JNDI Name** field, enter **LSR9_API.queue.AsyncDispatcher_error**.
The queue name and JNDI name must be the same.
 - h. Click **Next**.
 - i. From the **Subdeployments** list, select **cgJMSServer**.
 - j. Click **Finish**.
3. After you have created the **LSR9_API.queue.AsyncDispatcher** queue, you must associate it with the **LSR9_API.queue.AsyncDispatcher_error** queue:
- a. On the **Configuration** tab, under **Summary of Resources**, click **LSR9_API.queue.AsyncDispatcher**.
The Settings for LSR9_API.queue.AsyncDispatcher page appears.

- b. On the **Delivery Failure** tab, from the **Error Destination** list, select **LSR9_API.queue.AsyncDispatcher_error**.
- c. Click **Save**.

Creating Additional Queues on a Clustered Server

Creating additional queues on a clustered server involves the following:

- ◆ Create subdeployments for all the managed servers
- ◆ Create queues for all the managed servers
- ◆ Create distributed queues for all the managed servers

To create additional queues on a clustered server:

1. Create the the **LSR9_API.SubDeploy_n** subdeployment:
 - a. Log on to the WebLogic Server Administration Console.
 - b. In the Domain Structure tree, expand **Services**, expand **Messaging**, and then select **JMS Modules**.

The JMS Modules page appears.
 - c. Click **configwiz-jms**.

The Settings for configwiz-jms page appears.
 - d. On the **Subdeployments** tab, click **New**.
 - e. In the **Subdeployment Name** field, enter **LSR9_API.SubDeploy_1**, and then click **Next**.
 - f. Under JMS Servers, select the **cgJMSServer_auto_1** check box.
 - g. Click **Finish**.
 - h. On the **Subdeployments** tab, click **New**.
 - i. In the **Subdeployment Name** field, enter **LSR9_API.SubDeploy_2**, and then click **Next**.
 - j. Under JMS Servers, select the **cgJMSServer_auto_2** check box.
 - k. Click **Finish**
 - l. On the **Subdeployments** tab, click **New**.
 - m. In the **Subdeployment Name** field, enter **LSR9_API.SubDeploy_n**, and then click **Next**.
 - n. Under JMS Servers, select the **cgJMSServer_auto_n** check box.
 - o. Click **Finish**

2. Create the **LSR9_API.queue.AsyncDispatcher_auto_n** queue:
 - a. On the **Configuration** tab, under **Summary of Resources**, click **New**.

The Create a New JMS System Module Resource page appears.
 - b. Select **Queue** and click **Next**.
 - c. In the **Name** field, enter **LSR9_API.queue.AsyncDispatcher_auto_1**.
 - d. In the **JNDI Name** field, enter **LSR9_API.queue.AsyncDispatcher_auto_1**.

The queue name and JNDI name must be the same.
 - e. Click **Next**.
 - f. From the **Subdeployments** list, select **LSR9_API.SubDeploy_1**.
 - g. Click **Finish**.

Under JMS Servers, note that **LSR9_API.SubDeploy_1** is assigned to the JMS Server, **cgJMSServer_auto_1**.
 - h. On the **Configuration** tab, under **Summary of Resources**, click **New**.

The Create a New JMS System Module Resource page appears.
 - i. Select **Queue** and click **Next**.
 - j. In the **Name** field, enter **LSR9_API.queue.AsyncDispatcher_auto_2**.
 - k. In the **JNDI Name** field, enter **LSR9_API.queue.AsyncDispatcher_auto_2**.

The queue name and JNDI name must be the same.
 - l. Click **Next**.
 - m. From the **Subdeployments** list, select **LSR9_API.SubDeploy_2**.
 - n. Click **Finish**.

Under JMS Servers, note that **LSR9_API.SubDeploy_2** is assigned to the JMS Server, **cgJMSServer_auto_2**.
 - o. On the **Configuration** tab, under **Summary of Resources**, click **New**.

The Create a New JMS System Module Resource page appears.
 - p. Select **Queue** and click **Next**.
 - q. In the **Name** field, enter **LSR9_API.queue.AsyncDispatcher_auto_n**.
 - r. In the **JNDI Name** field, enter **LSR9_API.queue.AsyncDispatcher_auto_n**.

The queue name and JNDI name must be the same.
 - s. Click **Next**.
 - t. From the **Subdeployments** list, select **LSR9_API.SubDeploy_n**.

- u. Click **Finish**.
Under JMS Servers, note that **LSR9_API.SubDeploy_n** is assigned to the JMS Server, **cgJMSServer_auto_n**.
- 3. Create the **LSR9_API.queue.AsyncDispatcher_error_auto_n** queue:
 - a. On the **Configuration** tab, under **Summary of Resources**, click **New**.
The Create a New JMS System Module Resource page appears.
 - b. Select **Queue** and click **Next**.
 - c. In the **Name** field, enter **LSR9_API.queue.AsyncDispatcher_error_auto_1**.
 - d. In the **JNDI Name** field, enter **LSR9_API.queue.AsyncDispatcher_error_auto_1**.
The queue name and JNDI name must be the same.
 - e. Click **Next**.
 - f. From the **Subdeployments** list, select **LSR9_API.SubDeploy_1**.
 - g. Click **Finish**.
Under JMS Servers, note that **LSR9_API.SubDeploy_1** is assigned to the JMS Server, **cgJMSServer_auto_1**.
 - h. On the **Configuration** tab, under **Summary of Resources**, click **New**.
The Create a New JMS System Module Resource page appears.
 - i. Select **Queue** and click **Next**.
 - j. In the **Name** field, enter **LSR9_API.queue.AsyncDispatcher_error_auto_2**.
 - k. In the **JNDI Name** field, enter **LSR9_API.queue.AsyncDispatcher_error_auto_2**.
The queue name and JNDI name must be the same.
 - l. Click **Next**.
 - m. From the **Subdeployments** list, select **LSR9_API.SubDeploy_2**.
 - n. Click **Finish**.
Under JMS Servers, note that **LSR9_API.SubDeploy_2** is assigned to the JMS Server, **cgJMSServer_auto_2**.
 - o. On the **Configuration** tab, under **Summary of Resources**, click **New**.
The Create a New JMS System Module Resource page appears.
 - p. Select **Queue** and click **Next**.
 - q. In the **Name** field, enter **LSR9_API.queue.AsyncDispatcher_error_auto_n**.
 - r. In the **JNDI Name** field, enter **LSR9_API.queue.AsyncDispatcher_error_auto_n**.

The queue name and JNDI name must be the same.

- s. Click **Next**.
- t. From the **Subdeployments** list, select **LSR9_API.SubDeploy_n**.
- u. Click **Finish**.

Under JMS Servers, note that **LSR9_API.SubDeploy_n** is assigned to the JMS Server, **cgJMSServer_auto_n**.

- 4. After you have created the **LSR9_API.queue.AsyncDispatcher_auto_n** queues for all the managed servers, you must associate them with their respective **LSR9_API.queue.AsyncDispatcher_error_auto_n** error queues:

- a. On the **Configuration** tab, under **Summary of Resources**, click **LSR9_API.queue.AsyncDispatcher_auto_1**.

The Settings for LSR9_API.queue.AsyncDispatcher_auto_1 page appears

- b. On the **Delivery Failure** tab, from the **Error Destination** list, select **LSR9_API.queue.AsyncDispatcher_error_auto_1**.

- c. Click **Save**.

- d. On the **Configuration** tab, under **Summary of Resources**, click **LSR9_API.queue.AsyncDispatcher_auto_2**.

The Settings for LSR9_API.queue.AsyncDispatcher_auto_2 page appears.

- e. On the **Delivery Failure** tab, from the **Error Destination** list, select **LSR9_API.queue.AsyncDispatcher_error_auto_2**.

- f. Click **Save**.

- g. On the **Configuration** tab, under **Summary of Resources**, click **LSR9_API.queue.AsyncDispatcher_auto_n**.

The Settings for LSR9_API.queue.AsyncDispatcher_auto_n page appears.

- h. On the **Delivery Failure** tab, from the **Error Destination** list, select **LSR9_API.queue.AsyncDispatcher_error_auto_n**.

- i. Click **Save**.

- 5. Create the **LSR9_API.queue.AsyncDispatcher** distributed queue:

- a. On the **Configuration** tab, under **Summary of Resources**, click **New**.

The Create a New JMS System Module Resource page appears.

- b. Select **Distributed Queue** and click **Next**.

- c. In the **Name** field, enter **LSR9_API.queue.AsyncDispatcher**.

- d. In the **JNDI Name** field, enter **LSR9_API.queue.AsyncDispatcher**.

The queue name and JNDI name must be the same.

- e. Deselect the **Allocate Members Uniformly** check box.
- f. Click **Next**.
- g. From the **Available** list, select the **LSR9_API.queue.AsyncDispatcher** queue such as, **LSR9_API.queue.AsyncDispatcher_auto_1**, **LSR9_API.queue.AsyncDispatcher_auto_2**, and **LSR9_API.queue.AsyncDispatcher_auto_n**, and move it to the **Chosen** list by clicking the right arrow button.

The selected queue is moved to the **Chosen** list on the right. You can move a value from the **Chosen** list back to the **Available** list by clicking the left arrow button.
- h. Click **Finish**.

6. Create the **LSR9_API.queue.AsyncDispatcher_error** distributed queue:

- a. On the **Configuration** tab, under **Summary of Resources**, click **New**.

The Create a New JMS System Module Resource page appears.

- b. Select **Distributed Queue** and click **Next**.
- c. In the **Name** field, enter **LSR9_API.queue.AsyncDispatcher_error**.
- d. In the **JNDI Name** field, enter **LSR9_API.queue.AsyncDispatcher_error**.

The queue name and JNDI name must be the same.

- e. Deselect the **Allocate Members Uniformly** check box.
- f. Click **Next**.

From the **Available** list, select the **LSR9_API.queue.AsyncDispatcher_error** queue, such as **LSR9_API.queue.AsyncDispatcher_error_auto_1**, **LSR9_API.queue.AsyncDispatcher_error_auto_2**, and **LSR9_API.queue.AsyncDispatcher_error_auto_n**, and move it to the **Chosen** list by clicking the right arrow button.

The selected queue is moved to the **Chosen** list on the right. You can move a value from the **Chosen** list back to the **Available** list by clicking the left arrow button.

- g. Click **Finish**.

Deploying LSR Integration on the Application Server in a Single Server Environment

Deploying LSR integration on the application server in a single-server environment involves the following:

- ◆ Undeploying the existing LSR integration 9 application from the application server

◆ Deploying LSR integration 9.4 on the application server

To undeploy LSR integration 9 from the application server:

1. Start the WebLogic Server Administration Console using the following URL:

`http://ServerName:Port/console`

where:

ServerName is the name of the administration server.

Port is the administration server port number.

For example:

`http://wplsunsrvxmlapi:7001/console`

2. Enter the administration user name and password when prompted and press **Enter**.

3. Under **Change Center**, click **Lock & Edit**.

4. Expand the **Domain Structure** tree and click **Deployments**.

The Summary of Deployments window appears.

5. On the **Control** tab, select **lsr9_integration**.

6. From the **Stop** list, select **Force Stop Now**.

Ensure that the state of the LSR integration 9 application has changed from **Active** to **Prepared**.

7. Select **lsr9_integration** and click **Delete**.

The Delete Application Assistant window appears.

8. Click **Yes**.

The LSR integration 9 application is undeployed.

Before you deploy LSR integration 9.4, ensure the following servers are running:

◆ Administration server

If it is not running, start it using the following startup script:

For UNIX:

`domain_directory/startServerName.sh`

For Windows:

`domain_directory/startServerName.cmd`

where:

domain_directory is the WebLogic server domain directory.

ServerName is the name of the administration server.

To deploy LSR integration 9.4 on the application server:

1. Start the WebLogic Server Administration Console using the following URL:
`http://ServerName:Port/console`
where:
ServerName is the name of the administration server.
Port is the administration server port number.
For example:
`http://wplsunsvxmlapi:7001/console`
2. Enter the administration user name and password when prompted and press **Enter**.
3. Under **Change Center**, click **Lock & Edit**.
4. Expand the **Domain Structure** tree and click **Deployments**.
The Summary of Deployments window appears.
5. On the **Control** tab, click **Install**.
The Install Application Assistant window appears.
6. Under **Current Location**, navigate to the directory from which you want to deploy LSR integration 9.4 (for example, **opt/metasolv/mslv01/appserver/deploy**), select **lsr9_integration.ear**, and then click **Next**.
The Choose targeting style window appears.
7. Select **Install this deployment as an application** and click **Next**.
The Select modules to target window appears.
8. Select **cgJMSServer** and click **Next**.
The Select deployment targets window appears.
9. Under **JMSServers**, select **cgJMSServer**, and then click **Next**.
The Select modules to target window appears.
Ensure that **cgJMSServer** is displayed in the **Selected Targets** column.
10. Click **Next**.
11. Under **Source accessibility**, select **I will make the deployment accessible from the following location**.
12. Click **Finish**.
13. Under **Change Center**, click **Activate Changes**.
14. Expand the **Domain Structure** tree and click **Deployments**.

The Summary of Deployments window appears.

15. Select **lsr9_integration**.

16. From the **Start** list, select **Servicing all requests**.

The Start Application Assistant window appears.

17. Click **Yes**.

Ensure that the state of the LSR integration 9.4 application has changed from **Prepared** to **Active**.

18. If you receive the following error in the **appserver.mss.log** file during LSR deployment,

```
"log4j: ERROR Attempted to append to closed appender named
[XMLFileApp"
```

do the following:

a. Open the **loggingconfig.xml** file located in the *MSLV_Home/server/appserver/config* directory, where *server* is the name of the WebLogic server.

b. Remove the following entries:

```
category name="cmm.LSR"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLogger"
additivity="false"><level value ="error"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLevel"/>
<appender-ref ref="XMLFileApp"/>
</category>
<category name="LSR"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLogger"
additivity="false">
<level value="error"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLevel"/>
<appender-ref ref="XMLFileApp"/>
</category>
<category name="cmm.LSR"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLogger"
additivity="false">
<level value ="error"
class="com.metasolv.common.framework.logging.api.log4jext.MSLVLevel"/>
<appender-ref ref="XMLFileApp"/>
</category>
```

c. Save and close the file.

Deploying LSR Integration on the Application Server in a Clustered Environment

Deploying LSR integration on the application server in a clustered-server environment involves the following:

- ◆ Undeploying the existing LSR integration 9 application from the application server
- ◆ Deploying LSR integration 9.4 on the application server

To undeploy LSR integration 9 from the application server:

1. Start the WebLogic Server Administration Console using the following URL:

`http://ServerName:Port/console`

where:

ServerName is the name of the administration server.

Port is the administration server port number.

For example:

`http://wplsunsrvxmlapi:7001/console`

2. Enter the administration user name and password when prompted and press **Enter**.

3. Under **Change Center**, click **Lock & Edit**.

4. Expand the **Domain Structure** tree and click **Deployments**.

The Summary of Deployments window appears.

5. On the **Control** tab, select **cluster-lsr9_integration**.

6. From the **Stop** list, select **Force Stop Now**.

Ensure that the state of the LSR integration 9 application has changed from **Active** to **Prepared**.

7. Select **cluster-lsr9_integration** and click **Delete**.

The Delete Application Assistant window appears.

8. Click **Yes**.

The LSR integration 9 application is undeployed.

Before you deploy LSR integration 9.4, ensure the following servers are running:

- ◆ Administration server

If it is not running, start it using the following startup script:

For UNIX:

`domain_directory/startAdminServer.sh` (for non-XML deployment)

or

domain_directory/**startAdminServer_wli.sh** (for XML deployment)

For Windows:

domain_directory/**startAdminServer.cmd** (for non-XML deployment)

or

domain_directory/**startAdminServer_wli.cmd** (for XML deployment)

where:

domain_directory is the WebLogic server domain directory.

◆ Managed server or servers

Start any managed servers that are not running using the following startup script:

For UNIX:

domain_directory/**startServerName.sh**

For Windows:

domain_directory/**startServerName.cmd**

where:

domain_directory is the WebLogic server domain directory.

ServerName is the name of the administration server.

To deploy LSR integration 9.4 on the application server:

On Windows systems:

Enter the following at the command prompt:

```
java -cp
full_path\weblogic.jar;full_path\wlfullclient.jar;full_path\com.bea.core.des
criptor.wl_1.1.0.0.jar; weblogic.Deployer -adminurl t3://Host:Port -name
cluster-lsr9_integration -source MSLV_Home/deploy/lsr9_integration.ear -user
UserID -password Password -targets Cluster_Name -submoduletargets
cgJMSServer@WlwRuntimeAppScopedJMS@cgJMSServer_auto_1 -deploy -
usenonexclusivelock -verbose
```

On UNIX systems:

Enter the following at the command prompt:

```
java -cp
full_path\weblogic.jar:full_path\wlfullclient.jar:full_path\com.bea.core.des
criptor.wl_1.1.0.0.jar: weblogic.Deployer -adminurl t3://Host:Port -name
cluster-lsr9_integration -source MSLV_Home/deploy/lsr9_integration.ear -user
UserID -password Password -targets Cluster_Name -submoduletargets
cgJMSServer@WlwRuntimeAppScopedJMS@cgJMSServer_auto_1 -deploy -
usenonexclusivelock -verbose
```

where:

full_path is the path of the **lib** directory, which is contained in the directory where you extracted the MSS installer. For example, **opt/m62Single/single/installer/lib**.

Host is the name of the administration server.

Port is the administration server port number.

MSLV_Home is the directory in which the MetaSolv Solution software is installed.

UserID is the user name of the WebLogic server.

Password is the password of the WebLogic server.

Cluster_Name is the name of the target cluster.



Verify that these commands are run successfully.

About the LSR 9 Samples File

Oracle provides a sample application, `lsr9_samples`, as a reference and guideline for developing Workshop applications to interface with the LSR XML API. If you are a developer, the sample application is designed to help you understand and work with the XML APIs.

Where to Find the Sample File

During the installation, the **lsr9_samples.jar** file is stored on the application server in the *MSLV_Home/server/appserver/samples* directory, where *server* is the name of the WebLogic server.

lsr9_samples.jar—This file contains the code, libraries, and other files required for WebLogic 10.3.1 Workshop-based development for the `lsr9_samples` application.

lsr9_samples.ear—This is the representation of an EAR file that results from a successful build of the application. This file is contained in the **lsr9_samples.jar** file.

lsr9_samples_workspace.jar—This is the file that you must import into WebLogic Workshop. This file is contained in the **lsr9_samples.jar** file.

ReleaseNotes.doc—This file contains any release specific notes, including enhancements and fixes.

You can extract the contents of the JAR file to any location. For more information on setting up a samples file, see *XML API Developer's Reference*.

How You Can Use the Files

You can use the sample files in two ways:

- ◆ You can extract the files in the **lsr9_samples.jar** file, place them in a directory, and access them from WebLogic Workshop.
- ◆ You can deploy the **lsr9_samples.ear** file to observe an application. If you choose to deploy the application, you must create queues for the application just as you created them for LSR for use with the XML API option. See [“Creating Additional Queues on a Single Server”](#) for the procedure for creating the queues.

The queues must have the following names to work with the application in the **lsr9_samples.ear** file:

- ◆ TestWorkFlowsLSR9.queue.AsyncDispatcher
- ◆ TestWorkFlowsLSR9.queue.AsyncDispatcher_error