

Oracle® WebLogic Server

Creating WebLogic Domains Using the Configuration Wizard

10g Release 3 (10.1.3)

August 2008

ORACLE®

Copyright © 2007, 2008, 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

1. Introduction

Introduction to Domains	1-1
Overview of the Configuration Wizard	1-3
About Domain and Extension Templates	1-6
Creating a Domain	1-6
Extending a Domain.	1-8
Additional Tools for Creating, Extending, and Managing Domains	1-9

2. Starting the Configuration Wizard

Starting in Graphical Mode	2-1
Starting in Console Mode	2-2

3. Creating a WebLogic Domain

4. Customizing the Environment

Configuring the RDBMS Security Store Database	4-2
Configuring the Administration Server	4-3
Configuring Managed Servers	4-7
Configuring Clusters	4-7
Assigning Managed Servers to Clusters	4-9
Creating HTTP Proxy Applications	4-10
Configuring Machines	4-11
Assigning Servers to Machines	4-13

Review the Domain Settings and Create the Domain	4-14
--	------

5. Customizing JDBC and JMS Settings in WebLogic Domains

Configure JDBC Data Sources	5-1
Test Data Source Connections	5-5
Run Database Scripts	5-5
Configure JMS File Stores	5-6

6. Extending WebLogic Domains

Choose to Extend a Domain	6-2
Select a WebLogic Domain Directory	6-2
Select the Extension Source	6-2
Customize JDBC and JMS File Store Settings	6-2
Extend Your WebLogic Domain.	6-3

7. Examples: Using the Configuration Wizard

Creating a Domain	7-1
Extending an Existing Domain	7-2
Creating a Domain with Managed Servers and Clusters.	7-4

Introduction

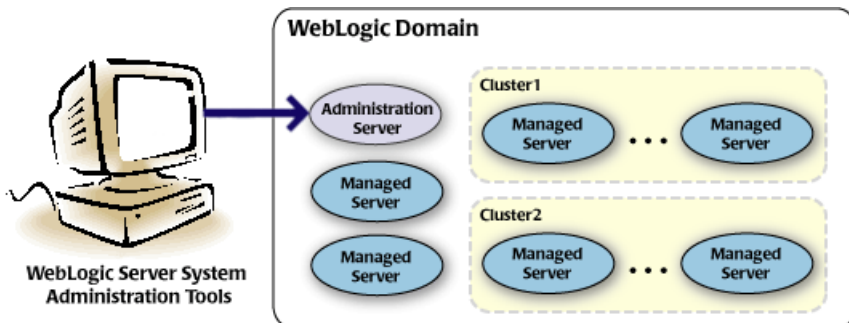
This guide provides information about configuring WebLogic Server domains by using the Configuration Wizard.

Introduction to Domains

Note: This section provides a brief introduction to domains. You can skip this section if you are familiar with the primary features of a domain.

A domain is the basic administration unit of WebLogic Server. It consists of one or more WebLogic Server instances, and logically related resources and services that are managed, collectively, as one unit.

Figure 1-1 WebLogic Domain Structure



As shown in [Figure 1-1](#), the basic domain infrastructure consists of one Administration Server and optional managed servers and clusters. These components are described in the following table.

Table 1-1 Domain Infrastructure Components

Feature	Description
Administration Server	<p>A domain includes one WebLogic Server instance that is configured as an administration server. The administration server provides a central point for managing the domain and providing access to the WebLogic Server administration tools. These tools include, but are not limited to, the following:</p> <ul style="list-style-type: none"> • WebLogic Server Administration Console: Graphical user interface to the Administration Server • WebLogic Server Node Manager: A Java program that enables you to start and stop server instances—both Administration Servers and Managed Servers—remotely, and to monitor and automatically restart them after an unexpected failure. <p>For more information about the WebLogic Server administration tools, see Summary of System Administration Tools and APIs in <i>Introduction to WebLogic Server and WebLogic Express</i>.</p>
Managed Servers	<p>All other WebLogic Server instances in a domain are called managed servers. Managed servers host application components and resources, which are also deployed and managed as part of the domain. In a domain with only a single WebLogic Server instance, that single server works as both the administration server and managed server.</p>
Clusters	<p>A domain may also include WebLogic Server clusters, which are groups of server instances that work together to provide scalability and high availability for applications. Clusters can improve performance and provide failover, should a server instance become unavailable. The servers within a cluster can either run on the same machine or reside in different machines. To the client, a cluster appears as a single WebLogic Server instance.</p>

Note: All managed servers in a domain must run the same version of WebLogic Server. The administration server can run either the same version as the managed servers in the domain, or a later service pack.

In addition to infrastructure components, a domain defines the basic network configuration for the server instances that it contains. Specifically, a domain defines application deployments, supported application services (such as database and messaging services), security options, and physical host machines.

Domain configuration information is stored in the configuration directories under the domain directory.

Common Domain Configurations

You might find it useful to configure multiple domains based on specific criteria, such as system administrator responsibilities, the logical classification of applications, the geographical locations of servers, or size. The following table outlines the most common domain configurations.

Table 1-2 Common Domain Configurations

Configuration	Description
Domain with managed servers	In typical production environments, several managed servers can host applications, and an administration server performs management operations.
Domain with managed servers and clusters	In production environments that require increased performance, throughput, or availability for an application, several managed servers might be grouped in a cluster. In such a case, the domain consists of one or more clusters with the applications they host, additional managed servers, if necessary, and an administration server to perform management operations.
Stand-alone Server Domain	In development or test environments, a single application and server might be deployed independently without managed servers. In such a case, you can have a domain consisting of a single administration server that also hosts the applications you want to test or develop.

Note: In production environments, it is recommended that you deploy applications only on managed servers; the administration server should be reserved for management tasks.

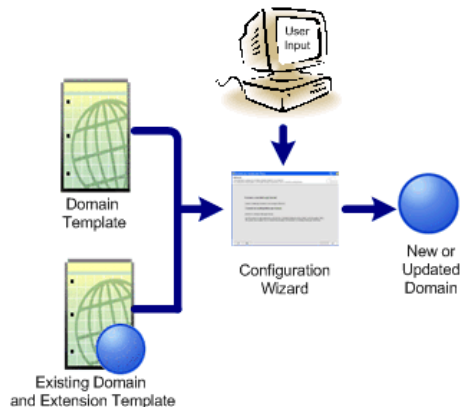
For more information about WebLogic Server domains, see “Understanding WebLogic Server Domains” in *Understanding Domain Configuration*.

Overview of the Configuration Wizard

Before you can develop and run a WebLogic application, you must first create a domain. The Configuration Wizard (illustrated in [Figure 1-2](#)), simplifies the process of creating and extending a domain. To create or extend a domain by using the Configuration Wizard, you simply select the product components to be included in the domain (or choose a template that best meets your

requirements), and provide basic configuration information. The Configuration Wizard then creates or extends the domain by using the settings from templates. For more information about templates, see [About Domain and Extension Templates](#).

Figure 1-2 Configuration Wizard



After you create a domain by using the Configuration Wizard, you can start a WebLogic Server instance to run in the domain for developing, testing, and deploying applications.

Note: The Domain Template Builder simplifies the process of creating templates by guiding you through the process of creating custom domain and extension templates. You can use these templates for creating and extending domains by using the Configuration Wizard or the WebLogic Scripting Tool (WLST).

For information about the Domain Template Builder, see [Creating Templates Using the Domain Template Builder](#).

For more information about the WLST, see “Creating Domains Using WLST” in [WebLogic Scripting Tool](#).

Modes of Operation

The Configuration Wizard can be used off-line only; that is, when there is no server running. It supports the following modes of operation:

- Graphical mode—an interactive, GUI-based mode
- Console mode—an interactive, text-based mode

Note: For a scripted, silent-mode method, you can use WLST. For more information, see [WebLogic Scripting Tool](#).

Configuration Wizard Output

A domain created using the Configuration Wizard has the following directories:

- `autodeploy`—provides a location from which you can deploy applications quickly on a development server. When the WebLogic Server instance is running in development mode, it automatically deploys any applications or modules that you place in this directory.
- `bin`—contains scripts to start and stop the administration server, and, optionally, managed servers.
- `config`—contains the following:
 - A domain-specific configuration file, `config.xml`, which specifies the name of the domain and the configuration parameter settings for each server instance, cluster, resource, and service in the domain.
 - Subdirectories that contain the configuration for various system modules: `deployments`, `diagnostics`, `jdbc`, `jms`, `lib`, `nodemanager`, and `security`. These subdirectories contain configuration files that are incorporated, by reference, into the `config.xml` file.

Note: Depending on your configuration, some subdirectories might not exist.
- `console-ext`—contains console extensions used by the administration server.
- `init-info`—contains files used by the Configuration Wizard to support creation and extension of the domain.
- `lib`—contains the domain library. When the server starts, any `jar` files that you place in this directory are added, dynamically, to the end of the server classpath.
- `security`—contains common security files for all the servers in the domain.
- `servers`—contains a subdirectory for each server in the domain. These server subdirectories, in turn, contain subdirectories that hold directories and files that must be different for each server in a domain, such as `bin`, `cache`, `data`, `logs`, `security`, and `tmp`.
- `user_staged_config`—if the domain is configured to be user-staged – that is, the administrator is responsible for staging (copying) the configuration information to the managed servers, this directory provides an alternative to the `config` directory.

If the template used to create a domain includes applications, the application files are located, by default, in `user_projects/applications/domain_name`.

For more information, see [Domain Configuration Files](#).

About Domain and Extension Templates

In the context of the Configuration Wizard, the term template refers to a Java Archive (JAR) file, which contains the files and scripts required to create or extend a domain. The types of templates that can be used by the Configuration Wizard to create or extend domains include:

- Domain template—defines the full set of resources within a domain, including infrastructure components, applications, services, security options, and general environment and operating system options. You can create this type of template from an existing domain by using the Domain Template Builder or the `pack` command. Subsequently, you can create a domain based on the template by using the Configuration Wizard

The product distribution includes a base WebLogic Server domain template. This template defines the core set of resources within a domain, including an administration server and basic configuration information, infrastructure components, and general environment and operating system options. It does not include sample applications. You can use this template to create a basic WebLogic Server domain, which you can then extend with applications and services, or additional product component functionality.

- Extension template—defines applications and services that can provide additional features, such as Apache Beehive, product sample applications, or JDBC or JMS components. This type of template can be used to update an existing domain.
- Managed server template—defines the subset (within a domain) of resources that are required to create a managed server domain on a remote machine. This type of template can be created using the `pack` command.

The product installation includes a set of predefined domain and extension templates. This set includes the base WebLogic Server domain template and various extension templates that allow you to add component features and samples to the base domain. For more information about these templates and how they relate to each other, see [Domain Template Reference](#).

Creating a Domain

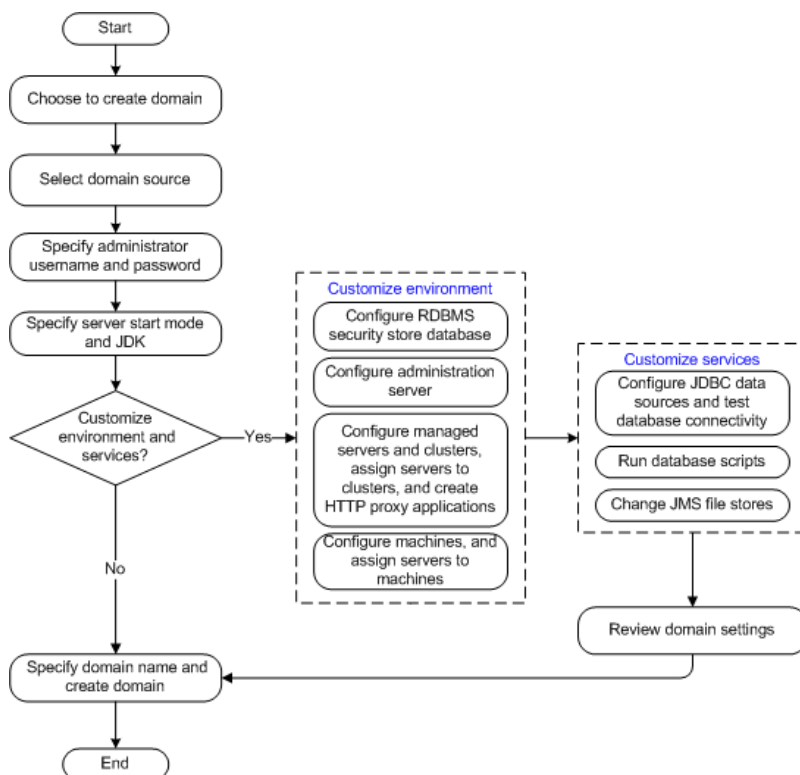
The Configuration Wizard guides you through the process of creating a domain for your target environment by selecting the product components that you want to include in your domain, or by using domain templates. If required, you can also customize the domain to suit your environment by adding and configuring managed servers, clusters, and machine definitions, or customizing predefined JDBC data sources, and JMS file store directories.

You might want to customize your domain in the following circumstances:

- To create a multi-server or clustered domain when using the default settings. All the predefined templates delivered with the product create single-server domains.
- To use a database that is different from the default database in the domain or extension template. In this case, you need to customize the JDBC settings to point to the appropriate database.
- To customize the listen port and the SSL port
- To create a test environment using a domain template that you received, and to modify the domain configuration to work in the test environment based on your requirement.

Figure 1-3 summarizes the steps for creating a domain by using the Configuration Wizard.

Figure 1-3 Creating a Domain



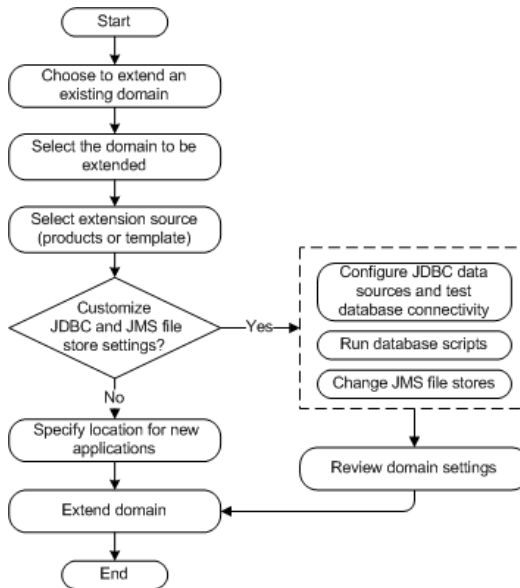
Extending a Domain

You can extend an existing domain by adding predefined applications and services, or additional product component functionality. For example, if you created a base WebLogic Server domain and want to add Workshop, you can extend the domain by using the Workshop extension.

To extend a domain by using the Configuration Wizard, select the domain that you want to extend and then select the additional product component. Alternatively, you can extend an existing domain by specifying an extension template to include additional applications and services. You can also customize the JDBC connections and change the JMS file store. The Configuration Wizard uses your input to update the configuration files, such as `config.xml`, and all other generated components in the domain directory, as required.

Figure 1-4 summarizes the steps for extending a domain by using the Configuration Wizard.

Figure 1-4 Extending a Domain



Additional Tools for Creating, Extending, and Managing Domains

As described earlier, you can create and extend domains by using the Configuration Wizard. In addition, you can use the tools listed in [Table 1-3](#) to create, extend, and manage domains. You can also perform run-time configuration by using the consoles of the product components.

Table 1-3 Additional Tools for Creating, Extending, and Managing Domains

To do this ...	Use the following tools...
Create a new domain or extend an existing domain	<ul style="list-style-type: none">• WebLogic Scripting Tool (WLST) WLST is a command-line scripting interface, which you can use to interact with and configure WebLogic Server instances and domains. When WLST is offline, it enables you to create a new domain or update an existing domain without connecting to a running WebLogic Server—supporting the same functionality as the Configuration Wizard. For more information, see WebLogic Scripting Tool.• <code>unpack</code> command You can use this command to create a domain from the command line, by using a template that is compatible with your current installation. You cannot use <code>unpack</code> to extend an existing domain. For more information, see Creating Templates and Domains Using the Pack and Unpack Commands.
Add applications and services, or modify existing settings	<ul style="list-style-type: none">• WebLogic Server administration console For more information, see WebLogic Server Administration Console Online Help.• Other system administration tools, such as WLST, <code>weblogic.Admin</code>, <code>JMX</code>, and <code>Ant</code>. For more information, see “Summary of System Administration Tools and APIs” in Introduction to WebLogic Server and WebLogic Express.
Manage and monitor the health and status of the domain	<ul style="list-style-type: none">• WebLogic diagnostic framework For more information, see Configuring and Using the WebLogic Diagnostics Framework.• WebLogic Server administration console For more information, see WebLogic Server Administration Console Online Help.• WebLogic Server node manager For more information, see Managing Server Startup and Shutdown.

Introduction

Starting the Configuration Wizard

This section describes how to start the Configuration Wizard in graphical and console modes.

- [Starting in Graphical Mode](#)
- [Starting in Console Mode](#)

Starting in Graphical Mode

When run in graphical mode, the Configuration Wizard is executed in a graphical environment.

The console for the machine on which the product installation resides must support Java-based GUIs. All Windows-based consoles support Java-based GUIs; only a subset of UNIX-based consoles support Java-based GUIs.

Note: If you attempt to start the Configuration Wizard in graphical mode on a system that cannot support graphical display, the Configuration Wizard automatically starts in console mode.

You can start the Configuration Wizard in graphical mode from either the Windows **Start** menu or from the command line.

- To start the Configuration Wizard in graphical mode on a Windows platform, choose **Start > Programs > Oracle WebLogic > WebLogic Server > Tools > Configuration Wizard**.
- To start the Configuration Wizard in graphical mode from a Windows command prompt or on a UNIX platform:
 - a. Log in to the system on which the product installation resides.

- b. Open an MS-DOS command prompt window (on Windows) or a command shell (on UNIX).
 - c. Go to the `\common\bin` subdirectory of the product installation directory.
 - d. Execute the following command:
On Windows: `config.cmd`
On UNIX: `sh config.sh`
- The Welcome window is displayed.

Starting in Console Mode

When run in console mode, the Configuration Wizard is executed in a text-based environment.

To start the Configuration Wizard in console mode:

1. Log in to the system on which the product installation resides.
2. Open an MS-DOS command prompt window (on Windows) or a command shell (on UNIX).
3. Go to the `\common\bin` subdirectory of the product installation directory.
4. Execute the following command:
 - Windows: `config.cmd -mode=console`
 - UNIX: `sh config.sh -mode=console`

Note: The command and arguments must be entered in lower case.

The Welcome screen is displayed.

To proceed, respond to the prompts by entering the number associated with your choice, pressing Enter. The arrow (->) adjacent to a choice indicates the current selection.

- To accept the current selection, type `next` (or `n`) and press Enter.
- To close the Configuration Wizard, enter `Exit` (or `x`) and press Enter.
- To review or change earlier selections, enter `Previous` (or `p`) and press Enter.

Creating a WebLogic Domain

Before you can develop and run WebLogic-based applications, you must first create a domain. The Configuration Wizard guides you through the process of creating a new domain quickly and easily, by selecting the product components that you want to include in your domain, or by using domain templates.

1. Start the Configuration Wizard as described in [Starting the Configuration Wizard](#).

The **Welcome** window is displayed.

2. Select **Create a new WebLogic domain** and click **Next**.

3. Select one of the following options:

- **Generate a domain configured automatically to support the following products:**

WebLogic Server is selected, by default. Select the check boxes corresponding to the other products that you want to include in the domain.

- **Base this domain on an existing template**

Select this option if you want to create a domain by using an existing domain template. For information about domain templates, see [Domain Template Reference](#).

Enter the full path to the template in the **Template location** field, or click **Browse** to navigate to the directory containing the required template.

Note: While using the Configuration Wizard or the WebLogic Scripting Tool to create or extend a clustered domain with a template that has applications containing application-scoped JDBC and/or JMS resources, you may need to perform additional steps (after the domain is created or extended) to make sure that the

application and its application-scoped resources are targeted and deployed properly in a clustered environment. For more information on targeting and deploying application-scoped modules, see [Deploying Applications and Modules with weblogic.deployer](#).

Click **Next**.

The **Configure Administrator Username and Password** window is displayed.

4. Configure the username and password for the administrator.
 - The username is used to boot the administration server and connect to it. Do not use commas or any of the following characters: \t, <, >, #, |, &, ?, (), { }.
 - The username is case sensitive.
 - The password value is encrypted. The password must contain at least eight characters, and is case sensitive.
 - Note:** Do not use the `weblogic` as the password in a production environment.

Click **Next**.

The **Configure Server Start Mode and JDK** window is displayed.

5. Select the WebLogic domain startup mode.
 - Development mode: In this mode, the configuration of security is relatively relaxed, allowing you to auto-deploy applications.
 - Production mode: In this mode, the configuration of security is stringent, requiring a user name and password to deploy applications. Before putting a domain into production, familiarize yourself with the securing the production environment. For more information, see [Securing a Production Environment](#).

For information about changing the run-time mode after you have created a domain, see [“Change to production mode”](#) in the *WebLogic Server Administration Console Online Help*.

[Table 3-1](#) provides information to help you choose a startup mode that suits your requirements.

Table 3-1 Differences Between Development Mode and Production Mode

Function	In development mode...	In production mode...
SSL	<p>You can use the demonstration digital certificates and the demonstration keystores provided by the WebLogic Server security services. With these certificates, you can design your application to work within environments secured by SSL.</p> <p>For more information about managing security, see Configuring SSL.</p>	<p>You must not use the demonstration digital certificates and the demonstration keystores. If you do so, a warning message is displayed.</p>
Deploying applications	<p>WebLogic Server instances can deploy and update applications that reside in the <code>domain_name/autodeploy</code> directory automatically.</p> <p>It is recommended that this method be used only in a single-server development environment.</p> <p>For more information, see Deploying Applications to WebLogic Server.</p>	<p>The auto-deployment feature is disabled; so, you must use the WebLogic Server administration console, the <code>weblogic.Deployer</code> tool, or the WebLogic Scripting Tool.</p>
Log file rotation	<p>By default, when you start the WebLogic Server instance, the server automatically renames (rotates) its local server log file as <code>SERVER-NAME.log.n</code>. For the remainder of the server session, messages accumulate in the log file until the file grows to a size of 500 kilobytes.</p> <p>For more information, see Rotate Log Files.</p>	<p>The server rotates the local log file after the size of the file reaches 5000 kilobytes.</p> <p>When the server is configured for production mode, by default, all versions of the log files are kept. Administrators may want to customize the number of log files retained.</p>
JDBC system resource	<p>The default capacity is 15 connections.</p>	<p>The default capacity is 25 connections.</p>

6. Select the JDK

In the **JDK Selection** pane, select the JDK for the startup mode that you selected in the **WebLogic Domain Startup Mode** pane.

The Configuration Wizard presents a list of the JDKs included in the installer. You can choose one of these JDKs or another JDK that you have installed on your system.

Note: If you select a JDK that is included in the installer, the Configuration Wizard creates server startup scripts to invoke the JDK. If you select a JDK that is not supplied by JDK, the Configuration Wizard does not configure the startup scripts; you must change the startup scripts manually. For more information about startup scripts, see [WLS Performance and Tuning](#).

Select only those JDKs that are supported on the platform you are using. For a list of the JDKs that are supported for a specific platform, see [Supported Configurations](#). The default selection reflects the JDK that best meets the requirements of your environment, based on the platform on which you are installing the domain.

Note: If you plan to use the JRockit JDK in production mode, it is recommended that you develop and test your applications by using JRockit early in the project cycle. For more information, see the [JRockit documentation](#).

Click **Next**.

The **Customize Environment and Services Settings** window is displayed.

7. While creating the domain, you can specify the RDBMS security store settings, configure the distribution of your domain across servers, clusters, and machines, specify JDBC data sources, define JMS file store settings.

- To retain the current settings, select **No** and click **Next**.

The **Create WebLogic Domain** window is displayed.

- To configure the distribution of your domain across servers, clusters, and machines, or to modify existing JDBC and JMS file store settings, select **Yes** and click **Next**.

The **Configure RDBMS Security Store Database** window is displayed.

For information about configuring the RDBMS security store, managed servers, clusters, and machines see [Customizing the Environment](#).

For information about specifying JDBC data sources and defining JMS file store settings, see [Customizing JDBC and JMS Settings in WebLogic Domains](#).

8. Enter the name of the domain and specify the domain location.
 - Domain names must not start with a number. This restriction prevents potential conflicts with internally-generated JDBC store table names, which must begin with a letter.
 - The domain directory can be located anywhere in the system. By default, it resides in `BEA_HOME\user_projects\domains\domain`, where `BEA_HOME` is the directory that

contains the product installation, and *domain* is the name of the domain directory defined by the selected template.

The Configuration Wizard stores the `config.xml` file and all other generated components in the domain directory that you specify.

9. Click **Create**.

Note: You cannot overwrite an existing domain. If a domain with the name you specified already exists in the selected location, you must either delete the existing domain, or specify a different name or location for the new domain.

The **Creating Domain** window displays status messages during the domain creation process.

When the process is complete, the new domain is ready for use.

- If you want to start the server immediately, select the **Start Admin Server** check box and click **Done**. This option is available only for Windows systems.
- If you do not want to start the server at this time, click **Done**.

Creating a WebLogic Domain

Customizing the Environment

While creating a domain, you can (optionally) specify the RDBMS security store settings, configure the distribution of your domain across servers, clusters, and machines, specify JDBC data sources, define JMS file store settings.

This section describes how you can configure the domain environment while creating a domain.

To configure the domain environment, you must select **Yes** in the **Customize Environment and Services Settings** window. The Configuration Wizard guides you through the following series of steps.

- [Configuring the RDBMS Security Store Database](#)
- [Configuring the Administration Server](#)
- [Configuring Managed Servers](#)
- [Configuring Clusters](#)
- [Assigning Managed Servers to Clusters](#)
- [Creating HTTP Proxy Applications](#)
- [Configuring Machines](#)
- [Assigning Servers to Machines](#)

Configuring the RDBMS Security Store Database

You can define RDBMS security store settings in the **Configure RDBMS Security Store Database** window, which is displayed when you select **Yes** in the **Customize Environment and Services Settings** window of the Configuration Wizard.

If RDBMS security store data already exists in `config.xml`, the data is displayed in read-only mode in the **Configure RDBMS Security Store Database** window.

- To retain the current settings, select **I don't want to change anything here**, and click **Next**.

Note: You can test the connection to the database by clicking **Test Connection**.

- To create or change the RDBMS security store settings, select **I want to create, change, or remove RDBMS support**. You can now enter values in the fields of the window.

[Table 4-1](#) describes the fields in the **Configure RDBMS Security Store Database** window.

Note: Fields marked with an asterisk are mandatory fields.

Table 4-1 Configure RDBMS Security Store Database

Field	Description
* Database Type	From the drop-down list, select the type of database that you want to use as the RDBMS security store. By default, the NONE option is selected. For information about supported databases, see Supported Configurations .
* Driver	Select the driver that you want to use for the database. The list of available drivers varies, depending on the database type that you select.
* Class Name	No action is required. The class name is displayed automatically based on the driver that you select.
* DBMS Name	Enter the name of the database.
* DBMS Host	Enter the name of the machine that hosts the database.
* DBMS Port	Enter the port to be used to connect to the server. The default port number that is associated with the selected database type is displayed automatically.

Table 4-1 Configure RDBMS Security Store Database

Field	Description
*URL	No action is required. The URL is displayed automatically based on the driver that you select.
*User Name	Enter the login name for connecting to the database.
*User Password	Enter the password for accessing the database. Valid values consist of a string of alphanumeric characters. The hyphen (-) and underscore (_) characters are supported. The value is encrypted.
*Confirm User Password	Re-enter the password.
*Known Properties	No action is required. The known properties of the database are displayed automatically based on the driver that you select.
Additional Properties	Enter additional properties, if any, to be passed to the driver.

After specifying the RDBMS security store settings, click **Next**.

Notes: You can test the connection to the database by clicking **Test Connection**.

Before starting the server, you must load the necessary SQL scripts for the RDBMS security store.

If you use an RDBMS security store in a clustered domain, it is recommended that you use it with JMS configuration (JNDI name and JMS topic). For more information, see the [WebLogic Server Administration Console Online Help](#).

Configuring the Administration Server

In every domain, one server must be designated as the administration server: the central point from which the whole domain is managed.

You can access the administrator server by using the URL `protocol://listen-address:listen-port`. The `protocol` can be any of the following: `t3`, `t3s`, `http`, `https`.

You can define the *listen-address* and *listen-port* in the **Configure the Administration Server** window of the Configuration Wizard.

Note: The **Configure the Administration Server** window is displayed when you select **Next** in the **Configure RDBMS Security Store Database** window of the Configuration Wizard.

[Table 4-2](#) describes the fields in the **Configure the Administration Server** window. Specify the appropriate values, and then click **Next** to proceed.

Note: Fields marked with an asterisk are required.

Table 4-2 Configuring the Administration Server

In this field . . .	Do the following . . .
Name*	<p>Enter a valid server name: a string of characters that can include spaces.</p> <p>Each server instance in the product environment must have a unique name, regardless of the domain or cluster in which it resides, and regardless of whether it is an administration server or a managed server. In addition, the name of the administration server must be unique among all component names within the domain.</p> <p>Note: This value is specified for identification purposes only; it is not used as part of the URL for applications that are deployed on the server. The server name is displayed in the WebLogic Server administration console. In addition, if you use WebLogic Server command-line utilities or APIs, you must specify this name to identify the server.</p>
Listen address	<p>From the drop-down list, select a value for the listen address.</p> <p>If you select localhost as the listen address for a server instance, non-local processes cannot connect to that server instance. Only processes on the machine that hosts the server instance can connect to the server instance. If the server instance must be accessible as localhost (for example, if you create administrative scripts that connect to localhost), and it must also be accessible by remote processes, select All Local Addresses. The server instance determines the address of the machine and listens on it.</p> <p>For more information, see Specifying the Listen Address.</p>
Listen port	<p>Enter a valid value for the listen port to be used for regular, non-secure requests (via protocols such as HTTP and T3). The default value is 7001. If you leave this field blank, the default value is used. The valid listen port range is from 1 to 65534.</p> <p>For more information, see Specifying the Listen Port.</p>

Table 4-2 Configuring the Administration Server

In this field . . .	Do the following . . .
SSL enabled	Select this check box to enable the SSL listen port. By default, SSL is disabled for all new servers.
SSL listen port	<p>This field is enabled only if the SSL enabled check box is selected.</p> <p>Enter a valid value to be used for secure requests (via protocols such as HTTPS and T3S). The default value is 7002. If you leave this field blank, the default value is used.</p> <p>The valid listen port range is from 1 to 65535.</p> <p>Note: By default, a server instance uses demonstration certificates to authenticate requests from a secure port. In a production environment, you must configure SSL to use certificates from a certificate authority. For more information, see Configuring SSL.</p> <p>For more information, see Specifying the Listen Port.</p>

Specifying the Listen Address

Table 4-3 provides guidelines for specifying the listen address for a server.

Table 4-3 Specifying Listen Address

If the listen address is set to . . .	Then ...
All Local Addresses or a DNS name	On multi-homed Windows machines, a server instance binds to all available IP addresses.
An IP address or a DNS name	<ul style="list-style-type: none"> • To connect to the server instance, processes can specify either the IP address or the corresponding DNS name. • Processes that specify localhost fail to connect. • You must update existing processes that use localhost to connect to the server instance. • For connections that specify the IP address for the listen address and a secured port for the listen port, host name verification must be disabled. <p>Note: To resolve a DNS name to an IP address, WebLogic Server must be able to contact an appropriate DNS server or obtain the IP address mapping locally. Therefore, if you specify a DNS name for the listen address, you must either leave a port open long enough for the WebLogic Server instance to connect to a DNS server and cache its mapping or you must specify the IP address mapping in a local file. If you specify an IP address for the listen address and then a client request specifies a DNS name, WebLogic Server attempts to resolve the DNS name, but if it cannot access DNS name mapping, the request fails.</p>
localhost	<ul style="list-style-type: none"> • Processes must specify localhost to connect to the server instance. • Only processes that reside on the machine that hosts the server instance (local processes) will be able to connect to the server instance.

Specifying the Listen Port

Note the following guidelines when specifying the listen ports and secure listen port:

- Although you can specify any valid port number, if you specify port 80, you can omit the port number from the HTTP request used to access resources over HTTP. For example, if you define port 80 as the listen port, you can use the URL:

`http://hostname/myfile.html` instead of
`http://hostname:portnumber/myfile.html`.

- On some operating systems, port 80 can be accessed only by processes run under a privileged user or group ID. In this case, you can assign the server instance to a UNIX machine on which a Post-Bind UID or GID is defined.
- In a development environment, you might want to run multiple instances of WebLogic Server on a single computer. If you do so, each instance must use a unique listen port-listen address combination.

On a multi-homed computer, you can use the same listen port but you must configure each server to use a unique IP address as the listen address. If your computer does not support multiple IP addresses, you must use a different listen port for each active instance.

Configuring Managed Servers

In production environments, enterprise applications are hosted, typically, on one or more managed servers, in addition to the administration server.

You can add and delete managed servers in the **Configure Managed Servers** window, which is displayed when you click **Next** in the **Configure the Administration Server** window of the Configuration Wizard.

Note: You can create managed servers on remote machines by using the `pack` and `unpack` commands. For more information, see [Creating and Starting a Managed Server on a Remote Machine](#).

1. Review the current managed server configurations. Default values may vary, based on the domain source you selected earlier.

Note: The wizard provides two views: a concise tabular view of all the managed servers and an individual view of each managed server, where each server is represented by a tab—you switch between servers by selecting the corresponding tab. To toggle the display mode between table and tab formats, click **Switch Display**.

2. Add or delete managed servers, or change the settings for existing managed servers, as required for your domain.
3. After configuring the managed servers, click **Next** to proceed.

Configuring Clusters

A cluster is a group of WebLogic Server instances that work together to provide scalability and high-availability for applications. By creating clusters, you can group managed servers such that they operate as a single unit for hosting applications and resources.

You can add, configure, and delete clusters in the **Configure Clusters** window of the Configuration Wizard. This window is displayed when you click **Next** in the **Configure Managed Servers** window, only if the domain contains at least one managed server.

1. Review the current cluster configuration. Default values may vary, based on the domain source you selected earlier.

Note: The wizard provides two views: a concise tabular view of all the clusters and an individual view of each cluster, where each cluster is represented by a tab—you switch between clusters by selecting the corresponding tab. To toggle the display mode between table and tab formats, click **Switch Display**.

2. Add or delete clusters, or change the settings for existing clusters, as required for your domain.

Note: Fields marked with an asterisk are required.

Table 4-4 Configuring Clusters

Field	Action
Name*	Enter a valid name for the cluster: a string of characters that can include spaces. The name of the cluster must be unique among all component names within the domain. The default value in this field is new_Cluster_n , where n is a numeric value that is used to differentiate among all the default cluster names; the value of n for the first cluster is 1. The value is incremented by 1 for each cluster that you add.
Multicast address	Enter the multicast address for the cluster. This address is used by cluster members to communicate with each other. The default value is 239.192.0.0. The valid multicast address range is 224.0.0.1 to 239.255.255.255.

Table 4-4 Configuring Clusters

Field	Action
Multicast port	<p>Enter the multicast port for the cluster.</p> <p>The multicast port is used by cluster members to communicate with each other. The default value is 7001.</p> <p>Valid values for multicast ports are from 1 to 65534.</p>
Cluster address	<p>Enter the addresses to identify the managed servers in the cluster.</p> <p>A cluster address can be one of the following:</p> <ul style="list-style-type: none">• Comma-separated list of IP addresses or DNS names and ports (for example: <code>dns_name:port, dns_name:port</code>)• DNS name that maps to multiple IP addresses• <code>localhost</code>, DNS name, or IP address if the listen address of all managed servers is listening to the same address with unique port numbers <p>The cluster address is used in entity and stateless EJBs to construct the host name portion of URLs. If the cluster address is not set, EJB handles may not work properly.</p>

3. After configuring the clusters, click **Next** to proceed.

Related Topics

[Setting Up WebLogic Clusters](#) in *Using WebLogic Server Clusters*

Assigning Managed Servers to Clusters

You can assign the available managed servers to clusters within the domain in the **Assign Servers to Clusters** window.

This window is displayed when you click **Next** in the **Configure Clusters** window, only if you have defined at least one cluster.

1. In the **Cluster** pane, select the cluster to which you want to assign a managed server.
2. Assign the managed server to the designated cluster in one of the following ways:
 - Double-click the name of the managed server in the **Server** pane.
 - Select the managed server and click the right arrow.
 - Shift+click to select multiple managed servers; then, click the right arrow.

The name of the managed server is removed from the **Server** pane and added below the name of the target cluster in the **Cluster** pane.

Note: Only managed servers are listed in the **Server** pane. The administration server is not listed because it cannot be assigned to a cluster.

3. Repeat steps 1 and 2 for each managed server that you want to assign to a cluster.
4. Review the cluster assignments.

If necessary, you can remove a managed server from a cluster in one of the following ways:

- Double-click the name of the managed server in the **Cluster** pane.
- Select the managed server and click the left arrow.

The name of the managed server is removed from the **Cluster** pane and restored to the **Server** pane.

5. Click **Next** to proceed.

Creating HTTP Proxy Applications

An HTTP proxy application acts as an intermediary for HTTP requests.

In the **Create HTTP Proxy Applications** window of the Configuration Wizard, you can create an HTTP proxy application for each cluster, and specify the managed server on which the proxy application must be deployed.

This window is displayed when you click **Next** in the **Assign Servers to Clusters** window, only if both of the following statements are true:

- At least one managed server is assigned to a cluster.
- At least one managed server is not assigned to any cluster.

To create HTTP proxy applications:

1. If multiple clusters are defined, select the tab corresponding to the cluster for which you want to create HTTP proxy applications.
2. Select the **Create HTTP proxy for cluster <cluster_name>** check box.

A list of the managed servers that are not assigned to any cluster is displayed in the **Proxy Server** drop-down list.

3. From the **Proxy Server** list, select a managed server on which the proxy applications must be deployed.

A proxy application named `BEAProxy4_clustername_servername` is created and targeted at the managed server.

4. Repeat steps 1 through 3 for each cluster for which you want to create HTTP proxy applications.
5. Click **Next** to proceed.

Configuring Machines

In a domain, the machine definitions identify physical units of hardware and are used to associate computers with the managed servers that they host.

You might want to create machine definitions in situations such as (but not limited to) the following:

- The administration server uses the machine definition, in conjunction with the node manager application, to start remote servers.
- WebLogic Server uses configured machine names when determining the server in a cluster that is best able to handle certain tasks, such as HTTP session replication. WebLogic Server then delegates those tasks to the identified server.

Note: You must configure machines for each product installation that runs a node manager process. The machine configuration must include values for the listen address and port number parameters.

You can create machine definitions in the **Configure Machines** window, which is displayed when you click **Next** in the **Create HTTP Proxy Applications** window of the Configuration Wizard.

1. Select the **Machine** tab (for Windows) and **Unix Machine** tab for (UNIX)
2. Review the current list of configurations, and add or change entries as required for your domain.
 - To add a machine, click **Add**.
 - To delete a machine, select the machine in the list and click **Delete**.

[Table 4-5](#) describes the configuration settings that you can define. Default values may vary based on the domain source that you selected earlier.

Note: Fields marked with an asterisk are required.

Table 4-5 Configure Windows Machine

Field	Description
Name*	<p>Enter a valid machine name: a string of characters that can include spaces.</p> <p>The machine name is used to identify the machine within the WebLogic Server domain; it is not required to match the network name for the machine.</p> <p>The name must be unique among all component names within the domain.</p> <p>The default value in this field is new_Machine_n, where n is a numeric value that is used to differentiate among all default machine names; the value of n for the first machine is 1. The value is incremented by 1 for each machine that you add.</p>
Node manager listen address	<p>Select a value from the drop-down list for the listen address used by node manager to listen for connection requests. By default, the IP addresses defined for the local system and localhost are shown in the drop-down list. The default value is localhost.</p> <p>If you specify an IP address for a machine that hosts the administration server and you need to access the WebLogic Server node manager, you must disable host name verification. For more information, see Using Host Name Verification in Securing WebLogic Server.</p>
Node manager listen port	<p>Enter a valid value for the listen port used by node manager to listen for connection requests.</p> <p>The valid node manager listen port range is from 1 to 65534.</p> <p>The default value is 5556.</p>
Post bind GID enabled	<p>This field is displayed only in the Unix Machine tab.</p> <p>Select this check box to enable a server running on this machine to bind to a UNIX group ID (GID) after it finishes all privileged startup actions (see Post bind GID). By default, this check box is not selected.</p>
Post bind GID	<p>This field is displayed only in the Unix Machine tab.</p> <p>Enter a valid UNIX group ID (GID) that a server running on this machine will run under after it finishes all privileged startup actions. Otherwise, the server continues to run under the group from which it was started. (Requires that you enable post-bind GID.)</p>

Table 4-5 Configure Windows Machine

Field	Description
Post bind UID enabled	This field is displayed only in the Unix Machine tab. Select this check box to enable a server running on this machine to bind to a UNIX user ID (UID) after it finishes all privileged startup actions (see Post bind UID). By default, this check box is not selected.
Post bind UID	This field is displayed only in the Unix Machine tab. Enter a valid UNIX user ID (UID) that a server running on this machine will run under after it finishes all privileged startup actions. Otherwise, the server continues to run under the account from which it was started. (Requires that you enable post-Bind UID.)

3. After updating the settings, click **Next**.

Assigning Servers to Machines

After configuring servers and defining machines, you can assign WebLogic Server instances to machines in the **Assign Servers to Machines** window.

This window is displayed only if you have defined at least one machine. It is displayed when you click **Next** in the **Configure Machines** window.

1. In the **Machine** pane, select the Windows or UNIX machine to which you want to assign a WebLogic Server instance.
2. Assign the appropriate WebLogic Server instance to the designated machine in one of the following ways:
 - Double-click the WebLogic Server instance in the **Server** pane.
 - Select the appropriate WebLogic Server instance in the **Server** pane and click the right arrow.
 - Shift+click to select multiple servers in the **Server** pane; then, click the right arrow.

The name of the WebLogic Server instance is removed from the **Server** pane and added, below the name of the target machine, in the **Machine** pane.

3. Repeat steps 1 and 2 for each WebLogic Server instance that you want to assign to a machine.

4. Review the machine assignments.

If necessary, you can remove a WebLogic Server instance from a machine in one of the following ways:

- Double-click the name of the appropriate WebLogic Server instance in the **Machine** pane.
- Select the appropriate WebLogic Server instance in the **Machine** pane and click the left arrow.

The name of the WebLogic Server instance is removed from the **Machine** pane and restored to the **Server** pane.

5. Click **Next**.

If the domain source on which you are basing your domain contains JDBC data source and JMS file store definitions, you are presented with the option to modify them as described in [Chapter 5, “Customizing JDBC and JMS Settings in WebLogic Domains.”](#) Otherwise, you are presented with the option to review the domain settings and create the domain.

Review the Domain Settings and Create the Domain

The **Review WebLogic Domain** window allows you to review the detailed configuration settings of your domain before the Configuration Wizard creates it.

1. Select an item in the **Domain Summary** pane on the left and review the associated details in the **Details** pane on the right. You can make limited adjustments by clicking **Previous** to return to a previous window.

Note: You can limit the type of information displayed in the **Domain Summary** pane by selecting a filter from the **Summary View** drop-down list.

2. After reviewing the domain settings, click **Next**.

The **Create WebLogic Domain** window is displayed

3. Enter the name of the domain and specify the domain location.

- Domain names must not start with a number. This restriction prevents potential conflicts with internally-generated JDBC store table names, which must begin with a letter.
- The domain directory can be located anywhere in the system. By default, it resides in `BEA_HOME\user_projects\domains\domain`, where `BEA_HOME` is the directory that

contains the product installation, and *domain* is the name of the domain directory defined by the selected template.

The Configuration Wizard stores the `config.xml` file and all other generated components in the domain directory that you specify.

4. Click **Create**.

Note: You cannot overwrite an existing domain. If a domain with the name you specified already exists in the selected location, you must either delete the existing domain, or specify a different name or location for the new domain.

The **Creating Domain** window displays status messages during the domain creation process.

When the process is complete, the new domain is ready for use.

- If you want to start the server immediately, select the **Start Admin Server** check box and click **Done**. This option is available only for Windows systems.
- If you do not want to start the server at this time, click **Done**.

Customizing the Environment

Customizing JDBC and JMS Settings in WebLogic Domains

When you are creating or extending a domain using the Configuration Wizard, you can change JDBC data source and JMS file store settings if they are defined in the domain or template that you selected as the source for domain that you are creating.

Note: If you select **Yes** in the **Customize Environment and Services Settings** window, the Configuration Wizard does not run any database automatically, even though domain JDBC is configured for PointBase database. In the **Run Database Scripts** dialog box, click **Run Scripts** screen, to load the respective database.

If you select **No** in the **Customize Environment and Services Settings** window, the Configuration Wizard automatically populates the required PointBase database.

The following topics describe the steps required to change the JDBC and JMS file store settings in your domain:

- [Configure JDBC Data Sources](#)
- [Test Data Source Connections](#)
- [Run Database Scripts](#)
- [Configure JMS File Stores](#)

Configure JDBC Data Sources

A JDBC data source contains a pool of database connections that are created when the data source instance is created—when it is deployed or targeted, or at server startup. Applications look up a

data source on the JNDI tree, and then request a connection. When the applications no longer needs the connection, they return the connections to the connection pool in the data source.

Note: No databases are defined in the `wls.jar` template; so, if you select the WebLogic Server template as the basis for the domain, the **Configure JDBC Data Sources** window is not displayed.

The **Configure JDBC Data Sources** window lets you configure the JDBC data sources defined in your domain source. This step is optional. If required, you can test the connections to the database using the **Test Connections** button.

Note: The values that you specify for your database in this window must match the actual configuration of the database. For example, if you change the name of the PointBase database for a preconfigured domain in the **DBMS name** field, a PointBase database configured with that name must exist.

1. Review the current list of JDBC data sources. Each data source is represented by a tab. You can view the information for a particular data source by selecting the corresponding tab. Default values may vary based on the domain source that you selected earlier.
2. Add or change entries as required for your domain. [Table 5-1](#) describes the fields of the **Configure JDBC Data Sources** window.

Note: Fields marked with an asterisk are required.

Table 5-1 Configure JDBC Data Sources

Field	Description
Name*	<p>Enter a valid name for the JDBC data source. The name must be a string of characters and can include spaces.</p> <p>The name of the JDBC data source must be unique among all component names within the domain.</p>
JNDI name*	<p>From the drop-down list, select a JNDI name to which this data source is bound.</p> <ul style="list-style-type: none"> • To add a new JNDI name, select Add New and enter a valid JNDI path. • To change an existing JNDI name, select the name and edit it. <p>You can associate multiple JNDI names with a single data source.</p> <p>When an application looks up a JNDI path, a <code>javax.sql.DataSource</code> instance corresponding to the data source is returned.</p>
Database type*	<p>From the drop-down list, select the type of database to which you want to connect. If your DBMS is not listed, select Other.</p>

Table 5-1 Configure JDBC Data Sources

Field	Description
Driver*	<p>From the drop-down list, select the JDBC driver that you want to use to connect to the database. The list includes common JDBC drivers for the selected DBMS.</p> <p>If you selected Other in the Database type field, this field is not available.</p>
Class name*	<p>If you selected a DBMS in the Database type field, no action is required.</p> <p>If you selected Other in the Database type field, enter the full package name of the class that implements the <code>java.sql.Driver</code> interface for your DBMS.</p>
DBMS name*	<p>Enter the name of the database.</p> <p>If you selected Other in the Database type field, this field is not available.</p>
DBMS host*	<p>Enter the name of the server machine hosting the database.</p> <p>If you selected Other in the Database type field, this field is not available.</p>
DBMS port*	<p>Enter the port to be used to connect to the server. The default setting associated with the database selected is displayed.</p> <p>If you selected Other in the Database type field, this field is not available.</p>
JDBC URL*	<p>If you selected a DBMS in the Database type field, no action is required. If the Driver name has been set and a default URL exists, that URL is displayed in this field.</p> <p>If you selected Other in the Database type field, enter the URL that should be used to create the connections in the connection pool in the data source.</p>
User name*	<p>Enter the account login name required for connecting to the database.</p> <p>If you selected Other in the Database type field, this field is not available. This value can be specified in the Additional Properties field.</p>
User password*	<p>Enter a password that is valid for accessing the database. Valid values consist of a string of alphanumeric characters. The hyphen (-) and underscore (_) characters are supported.</p> <p>This password overrides the password entered as part of the JDBC Properties.</p> <p>The value is encrypted.</p>
Confirm user password*	<p>Re-enter the user password.</p>

Table 5-1 Configure JDBC Data Sources

Field	Description
Known properties	<p>If you selected a DBMS in the Database type field, no action is required. This field displays the properties list that is passed to the JDBC driver for creation of physical database connections.</p> <p>If you selected Other in the Database type field, this field is blank and not available.</p>
Additional properties	<p>Enter any additional properties to be passed to the JDBC driver.</p> <p>If you specified Other in the Database type field, enter any properties to be passed to the JDBC driver, such as the property needed to specify the user.</p>
Supports global transactions	<p>If you selected an XA driver in the Driver field, Supports global transactions and Two-phase commit protocol are selected automatically. You cannot change the protocol.</p> <p>If you selected a non-XA driver in the Driver field, you can, if required, select Supports global transactions. Then, you can select one of the following protocols:</p> <ul style="list-style-type: none"> • Logging Last Resource—With this option, the transaction branch in which the connection is used is processed as the last resource in the transaction and is processed as a one-phase commit operation. The result of the operation is written in a log file on the resource itself, and the result determines the success or failure of the prepare phase of the transaction. This option offers some performance benefits with greater data safety than Emulate Two-Phase Commit, but it has some limitations. For more information, see Understanding the Logging Last Resource Transaction Option in <i>Configuring and Managing WebLogic JDBC</i>. • Emulate two-phase commit—With this option, the transaction branch in which the connection is used always returns success for the prepare phase of the transaction. It offers performance benefits, but also has risks to data in some failure conditions. Select this option only if your application can tolerate heuristic conditions. For more information, see Understanding the Emulate Two-Phase Commit Transaction Option in <i>Configuring and Managing WebLogic JDBC</i>. • One phase commit (default)—With this option, a connection from the data source can be the only participant in the global transaction and the transaction is completed by using a one-phase commit optimization. If more than one resource participates in the transaction, an exception is thrown.

3. After updating the settings, do one of the following:

- If you want to test the data source connections to the specified databases, click **Test Connections**. The **Test Data Source Connections** window is displayed. For more information, see [Test Data Source Connections](#).

- If you do not want to test the connections to the database, click **Next**.

Test Data Source Connections

The **Test Data Source Connections** window allows you to test the connection to the database for each of the data sources defined in your domain, by using the JDBC URL defined for the database. This window is only displayed if you clicked **Test Connections** on the **Configure JDBC Data Sources** window.

A list of the data sources and the associated JDBC URLs is displayed.

1. Make sure that the database to which you want to test the connections is running.
2. Click **Test** for the connection that you want to test.

The button changes to **Cancel**. You can press **Cancel** at any time to cancel the test. Once a test is in progress, all the other **Test** buttons are disabled.

The **Status** field is blank for connections for which the test has not been initiated.

3. Review the results of the test in the **Connection Result Log** pane.
4. Perform the test for each data source, as required.
5. Click **OK** to return to the **Configure JDBC Data Sources** window.

Run Database Scripts

A domain template might contain a set of SQL files organized by database type. If the domain template contains SQL files, you can run them while creating the domain, in the **Run Database Scripts** window. Database content for each of the data sources defined in your domain is set up by using pre-existing SQL or database loading files.

Notes: No databases are defined in the `wls.jar` template; so, if you selected the WebLogic Server template as the basis for the domain, the **Configure JDBC Data Sources** window and the **Run Database Scripts** window are not displayed.

When you use PointBase in production mode, it is mandatory to run the database scripts. In development mode, this step is not mandatory.

If you use the default PointBase database, you do not need to start the database server before running the scripts. For all other databases, the database server must be running to execute the SQL scripts.

1. In the **Available JDBC Data Sources** pane, select the data source for which you want to run the scripts. The scripts that can be executed are displayed in the **Available SQL Files and Database Loading Options**.
2. Select the database version from the **DB Version** drop-down menu.
3. Click **Run Scripts**.

All the scripts displayed in the **Available SQL Files and Database Loading Options** pane for the selected data source are executed, and the results are displayed in the **Results** pane. If you want to capture test output in a log file, select the **Log File** check box and specify the location of the log file.

4. Repeat steps 1 through 3 for each data source for which you want to execute SQL scripts.
5. Click **Next**.

Configure JMS File Stores

A JMS file store is a disk-based file in which persistent messages can be saved.

You can modify the JMS file stores that are configured in your domain, in the **Configure JMS File Stores** window, which is displayed when you click **Next** in the **Run Database Scripts** window. This step is optional.

1. Review the current list of JMS file stores. Default values may vary based on the domain source that you selected earlier.

Note: The wizard provides two display modes: a concise tabular view of all the defined components, and an individual view, in which each component is represented by a tab, and you view a particular component by selecting the corresponding tab. To toggle the display mode between table and tab formats, click **Switch Display**.

2. Modify the settings, as required for your domain.

Table 5-2 describes the fields of the **Configure JMS File Stores** window.

Table 5-2 Configure JMS File Stores

Field	Description
Name*	<p>Enter a name for the JMS file store. The name must be a string of characters and can include spaces.</p> <p>The name of the JMS file store must be unique among all component names within the domain.</p>
Directory	<p>Enter the path of the directory (in your system) in which the JMS file store resides.</p>
Synchronous write policy	<p>From the drop-down list, select one of the following synchronous write policies to determine how the file store writes data to the disk:</p> <ul style="list-style-type: none">• Cache-Flush: Transactions cannot be completed until all their write operations have been flushed to the disk.• Direct-Write: Write operations are performed directly to the disk. This policy is supported on Solaris and Windows. If this policy is active on an unsupported platform, the file store switches automatically to the cache-flush policy.• Disabled: Transactions are complete as soon as the writes are cached in memory. When this policy is active, completion of transactions does not depend on waiting for writes to reach the disk. <p>This setting affects performance, scalability, and reliability.</p> <p>Note: The use of the direct-write policy is transactionally reliable in Solaris systems, but Windows systems may leave transaction data in the on-disk cache without writing it to disk immediately. This is not considered to be transactionally reliable because a power failure can cause loss of on-disk cache data, possibly resulting in lost and/or duplicate messages. For reliable writes using direct-write on Windows, either disable all write caching for the disk (enabled by default), or use a disk with a battery-backed cache. Some file systems, however, do not allow this value to be changed (for example, a RAID system that has a reliable cache).</p> <p>Note: If the JMS file store is used exclusively for paging non-persistent messages to the disk, the synchronous write policy is ignored.</p>

3. After updating the settings, click **Next**.

Related Topics

[Creating a Custom \(User-Defined\) File Store](#) in *Configuring WebLogic Server Environments*

Extending WebLogic Domains

You can add product component functionality, or additional applications and services to an existing domain by extending it using the Configuration Wizard. For example, if you created a base WebLogic Server domain and you want to add the Avitek Medical Records Examples functionality, you must extend the domain by using the `medrec.jar` template.

The Configuration Wizard simplifies the task of extending an existing domain by using extension templates. For more information about domain templates, see [Domain Template Reference](#).

The following topics describe the steps required to extend an existing domain by using the Configuration Wizard:

- [Choose to Extend a Domain](#)
- [Select a WebLogic Domain Directory](#)
- [Select the Extension Source](#)
- [Customize JDBC and JMS File Store Settings](#)
- [Extend Your WebLogic Domain](#)

Note: You cannot extend an active domain by using the Configuration Wizard. Before you proceed with extending a domain, make sure that the servers in the domain are not running.


Choose to Extend a Domain

The **Welcome** window of the Configuration Wizard prompts you to choose whether you want to create a new domain or extend an existing domain by adding product component functionality, applications and services.

Select **Extend an existing WebLogic domain** and click **Next**.

Select a WebLogic Domain Directory

The **Select a WebLogic Domain Directory** window prompts you to select the existing domain that you want to update with additional applications or services.

1. Use the navigation tree to select a valid domain directory (a directory that contains a **config.xml** file in the **config** directory of the domain. A valid domain directory is indicated by the  icon.
2. Click **Next**.

Select the Extension Source

The **Select Extension Source** window prompts you to select the source from which to extend your domain. You can select products to add to your domain, or extend your domain by using an existing extension template.

1. Choose one of the following options:
 - **Extend my domain automatically to support the following added products:**

Select the check boxes for the products that you want to add to your domain. The products already included in your domain are indicated by a grayed-out check box.
 - **Extend my domain using an existing extension template**

Specify the path to the extension template in the **Template location** field and click **Next**.
2. Click **Next**.

Customize JDBC and JMS File Store Settings

The **Customize JDBC and JMS File Store Settings** window lets you modify existing JDBC and JMS file store settings. If you choose not to customize any JDBC or JMS settings by accepting the default (**No**), you proceed directly to extending the domain.

For information about customizing JDBC data sources and JMS file stores, see [Configure JDBC Data Sources](#) and [Configure JMS File Stores](#).

Extend Your WebLogic Domain

The **Extend WebLogic Domain** window displays the name and location for your domain.

If the domain contains an applications directory and the domain extension does not contain application files, the location for that directory is also displayed.

1. If the domain extension contains application files, specify the path to the applications directory in the **Applications location** field.
2. Click **Extend** to extend the domain with the applications, services, and settings based on the specified extension template.

The **Creating Domain** window displays status messages during the domain extension process. The Configuration Wizard updates the **config.xml** file and other application-specific components in the domain directory, as defined by the domain template. When the process is complete, the updated domain is ready for use.

3. Click **Done**.

Extending WebLogic Domains

Examples: Using the Configuration Wizard

This section provides information and examples for common domain configuration tasks that you can perform by using the Configuration Wizard:

- [Creating a Domain](#)
- [Extending an Existing Domain](#)
- [Creating a Domain with Managed Servers and Clusters](#)

Creating a Domain

The following example describes how to create a WebLogic Server domain.

1. Start the Configuration Wizard in graphical mode.
 - On Windows, choose **Start > Programs > Oracle WebLogic > WebLogic Server > Tools > Configuration Wizard**
 - On UNIX, go to the `/common/bin` subdirectory of the product installation directory, and enter `sh config.sh`.
2. In the **Welcome** window, select **Create a new WebLogic domain** and click **Next**.
3. In the **Select a Domain Source** window:
 - a. Select **Generate a domain configured automatically to support the following products**.

Note that the **WebLogic Server** check box is selected by default because it is a prerequisite for all WebLogic domains.

- b. Click **Next**.
4. In the **Configure Administrator Username and Password** window, enter a valid username and password, and click **Next**. This username is used to boot the administration server and connect to it.
5. In the **Configure Server Start Mode and JDK** window, specify whether to start the server in development mode or production mode, and select the JDK. Then, click **Next**.
6. In the **Customize Environment and Services Settings** window, indicate whether you want to change the distribution of your domain across servers, clusters, and machines, or modify existing JDBC and JMS file store services.
 - To add managed servers and clusters to your domain, or change JDBC and JMS settings, select **Yes** and click **Next**.
 - To accept the default settings, leave **No** selected and click **Next**.

In this example, we are not customizing the domain. Leave **No** selected and click **Next**.

7. In the **Create WebLogic Domain** window, enter a name for the domain and specify a location for the domain directory.

For example, enter `my_default_domain` as the name.

By default, the domain is created in `BEA_HOME\user_projects\domains\domain`, where `BEA_HOME` is the directory that contains the product installation, and `domain` is the name that you specified (`my_default_domain` in this example).


Click **Create**.

8. In the **Creating Domain** window, after the domain has been created, do one of the following:
 - If you want to start the server immediately, select the **Start Admin Server** check box and click **Done**. This option is available only for Windows systems.
 - If you do not want to start the server at this time, click **Done**.

Extending an Existing Domain

This example illustrates the procedure to extend the WebLogic Server domain that you created in [Creating a Domain](#), by using the Workshop extension template.

Note: Make sure that the servers in the domain are not running; you cannot update an active domain.

1. Start the Configuration Wizard in graphical mode.
 - On Windows, choose **Start > Programs > Oracle WebLogic > WebLogic Server > Tools > Configuration Wizard**
 - On UNIX, go to the `/common/bin` subdirectory of the product installation directory, and enter `sh config.sh`.
2. In the **Welcome** window, choose **Extend an existing WebLogic domain** and click **Next**.
3. In the **Select a WebLogic Domain Directory** window, navigate to the domain directory that you want to extend. Valid domain directories are indicated by the  icon. Select the domain to be extended and click **Next**.

For this example, navigate to `BEA_HOME\user_projects\domains` and select `my_default_domain`.

4. In the **Select Extension Source** window, you can choose to extend your domain by selecting products or an extension template.

To extend your domain by using the Workshop extension template:

 - a. Select **Extend my domain automatically to support the following added products**.
 - b. Select the **Workshop for WebLogic** check box and click **Next**.
5. In the **Customize JDBC and JMS Settings** window, you have the option to customize JDBC and JMS file store settings if they were previously defined in the domain or in the extension template.
 - To customize the JDBC and JMS file store settings, select **Yes** and click **Next**.
 - To accept the default settings, leave **No** selected and click **Next**.

For this example, leave **No** selected and click **Next**.

6. In the **Extend WebLogic Domain** window, you can specify an applications directory to contain the application files that are being added to the domain.

Click **Extend**.

7. In the **Creating Domain** window, click **Done** to close the Configuration Wizard.

Creating a Domain with Managed Servers and Clusters

This example describes the procedure to create a Workshop domain that is based on the basic WebLogic Server domain template, and to customize the domain to include the following:

- Two managed servers

A managed server hosts application components and resources, which are deployed and managed as part of the domain.

The two managed servers are in addition to the administration server.

- A single cluster

A cluster is a group of WebLogic Server instances that work together to provide scalability and high availability for applications. Clusters can improve performance and provide failover if a server instance becomes unavailable. The servers within a cluster can run on the same machine or on different machines. To the client, a cluster appears as a single WebLogic Server instance.

In this example, all the servers and the cluster created by the Configuration Wizard are intended to run on your local machine.

1. Start the Configuration Wizard in graphical mode.
 - On Windows, choose **Start > Programs > Oracle WebLogic > WebLogic Server > Tools > Configuration Wizard**
 - On UNIX, go to the `/common/bin` subdirectory of the product installation directory, and enter `sh config.sh`.
2. In the **Welcome** window, select **Create a new WebLogic domain** and click **Next**.
3. In the **Select a Domain Source** window:
 - a. Select **Generate a domain configured automatically to support the following products**.

Note that the **WebLogic Server** check box is selected by default since it is a prerequisite for all WebLogic domains.
 - b. Select the **Workshop for WebLogic** check box and click **Next**.
4. In the **Configure Administrator Username and Password** window, enter a valid username and password, and click **Next**. This username is used to boot the administration server and connect to it.

5. In the **Configure Server Start Mode and JDK** window, specify whether to start the server in development mode or production mode, and select the JDK.
6. Click **Next**.
7. In the **Customize Environment and Services Settings** window, select **Yes** and click **Next**.
8. In the **Configure RDBMS Security Store Database** window, click **Next**.
9. In the **Configure the Administration Server** window,
 - a. enter the following values:
 - **Name:** `MyAdminServer`
 - **Listen address:** `127.0.0.1`
 - **Listen port:** `7001`
 - **SSL listen port:** `7002`
 - b. Select the **SSL enabled** check box.
 - c. Click **Next**.
10. In the **Configure Managed Servers** window, add two managed servers **MS1** and **MS2**:
 - a. Click **Add**, and configure the first managed server with the following details:
 - Name:** `MS1`
 - Listen address:** `127.0.0.1`
 - Listen port:** `8001`
 - SSL listen port:** `8011`
 - b. Click **Add**, and configure the second managed server with the following details:
 - Name:** `MS2`
 - Listen address:** `127.0.0.1`
 - Listen port:** `8101`
 - SSL listen port:** `8111`
 - c. Select the **SSL enabled** check box.
 - d. Click **Next**.

11. In the **Configure Clusters** window, configure a cluster named **MY_CLUSTER**:

- a. Click **Add** and configure the cluster with the following details:

Name: **MY_CLUSTER**

Multicast address: **239.192.0.0**

Multicast port: **8050**

Cluster address: **127.0.0.1**

Note: All the managed servers listen at the same IP address (relying on unique port numbers to keep them separate); so you should set the cluster address to the same IP address as that of the managed servers.

- b. Click **Next**.

12. In the **Assign Servers to Clusters** window, assign the MS1 and MS2 managed servers to the **MY_CLUSTER** cluster.

- a. In the **Server** pane, click **MS1**. Then, shift+click **MS2**.

- b. Click the right arrow that is situated in between the **Server** and **Cluster** panes.

The names of the managed servers are moved from the **Server** pane to the **Cluster** pane.

- c. Click **Next**.

13. In the **Configure Machines** window, you define the configuration information for the Windows and UNIX machines in the domain. This step is optional.

Note: In this example, all the servers in the domain reside on your local machine, so it is not necessary to set up additional machines and assign target servers to them.

Click **Next**.

14. In the **Configure JDBC Data Sources** window, leave the default settings, and click **Next**.

15. In the **Configure JMS File Stores** window, leave the default settings, and click **Next**.

16. In the **Review WebLogic Domain** window, review the detailed configuration settings of your domain before the Configuration Wizard creates it.

The **Domain Summary** pane shows the **Deployment** view by default. By selecting different options from the **Summary View** drop-down list, you can see different views of the domain summary.

Select an item in the **Domain Summary** pane on the left and review the associated details in the **Details** pane on the right. You can make limited adjustments by clicking **Previous** to return to a previous window.

After reviewing the contents of your domain, click **Next**.

17. In the **Create WebLogic Domain** window, enter a name for the domain and specify a location for the domain directory.

For this example, enter **myclusterdomain** as the name of the domain.

By default, the domain is created in *BEA_HOME\user_projects\domains\domain*, where *BEA_HOME* is the directory that contains the product installation, and *domain* is the domain name that you specify (*myclusterdomain* in this example).

Click **Create**.

18. In the **Creating Domain** window, after the domain has been created, do one of the following:

- If you want to start the server immediately, select the **Start Admin Server** check box and click **Done**. This option is available only for Windows systems.
- If you do not want to start the server at this time, click **Done**.

Examples: Using the Configuration Wizard